

ИЗ ФОНДОВ РОССИЙСКОЙ ГОСУДАРСТВЕННОЙ БИБЛИОТЕКИ

Слинкин, Дмитрий Анатольевич

1. Использование метода проектов при обучении
программированию в курсе информатики

1.1. Российская государственная библиотека

Слинкин, Дмитрий Анатольевич

Использование метода проектов при обучении программированию в курсе информатики [Электронный ресурс]: Дис. канд. пед. наук : 13.00.02 .-М.: РГБ, 2003 (Из фондов Российской Государственной библиотеки)

Теория и методика обучения и воспитания (по областям и уровням образования)

Полный текст:

<http://diss.rsl.ru/diss/03/0166/030166008.pdf>

Текст воспроизводится по экземпляру,
находящемуся в фонде РГБ:

Слинкин, Дмитрий Анатольевич

Использование метода проектов при обучении
программированию в курсе информатики

Екатеринбург 2001

Российская государственная библиотека, 2003
год (электронный текст).

64:01-13/1973-4

Шадринский государственный педагогический институт

На правах рукописи

Слинкин Дмитрий Анатольевич

**ИСПОЛЬЗОВАНИЕ МЕТОДА ПРОЕКТОВ ПРИ ОБУЧЕНИИ
ПРОГРАММИРОВАНИЮ В КУРСЕ ИНФОРМАТИКИ**

13.00.02 – теория и методика обучения и воспитания
(информатика, уровень общего образования)

Диссертация на соискание ученой степени
кандидата педагогических наук

Научные руководители:

доктор технических наук,
профессор Красноперов Г.В.

доктор педагогических наук,
профессор Шамало Т.Н.



Екатеринбург 2001

ОГЛАВЛЕНИЕ

Введение.....	4
ГЛАВА 1. ИСПОЛЬЗОВАНИЕ МЕТОДА ПРОЕКТОВ В УЧЕБНОМ ПРОЦЕССЕ	13
1.1. Метод проектов как педагогическая технология.....	13
1.2. Типология проектов, используемых образовательном процессе.....	30
1.3. Организация технологического процесса обучения программированию с использованием метода проектов	46
ГЛАВА 2. МЕТОДИКА ИСПОЛЬЗОВАНИЯ МЕТОДА ПРОЕКТОВ ПРИ ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ В СТАРШИХ КЛАССАХ ОБЩЕ-ОБРАЗОВАТЕЛЬНОЙ ШКОЛЫ	54
2.1. Роль метода проектов в развитии познавательных потребностей школьников в процессе обучения программированию	54
2.2. Принципы построения спецификаций программных проектов в общеобразовательной школе	73
2.3. Методика обучения программированию в 10-11 классах общеобразовательной школы на основе использования метода проектов	84
2.4. Подготовка будущих учителей информатики к использованию метода проектов	102

ГЛАВА 3. МЕТОДИКА ПРОВЕДЕНИЯ И РЕЗУЛЬТАТЫ ПЕДАГОГИЧЕСКОГО ЭКСПЕРИМЕНТА	116
3.1. Задачи и этапы обучающего эксперимента	116
3.2. Содержание и результаты педагогического эксперимента.....	123
Заключение	139
Библиографический список	141
Приложения	158

ВВЕДЕНИЕ

Актуальность исследования. Социальные и экономические изменения в России, быстрый технический прогресс, информатизация общества ставят новые цели перед образованием, одной из которых является формирование творческого мышления и продуктивной созидательной деятельности учащегося как условие его самореализации в жизни.

Новые информационные технологии изменили деятельность человека практически во всех областях, обеспечив получение эффективного доступа к неограниченным информационным ресурсам, возможность накопления и обмена данными. Доступность и объем специализированной информации выводит на первый план способность легко ориентироваться в современном технологическом обществе, самостоятельно анализировать проблемы, выявлять перспективные цели и планировать оптимальные пути их достижения, воплощать независимо принятые решения на практике и оценивать их последствия и результаты. Эти условия отразились и на образовательной деятельности как с точки зрения содержания задач обучения, так и в отношении использования открывшихся технологических возможностей для достижения учебных целей.

Решения проблем, связанных с развитием у школьников умений и навыков самообразования, самостоятельности, обучением творческому подходу к поставленной задаче возлагаются на общую дидактику и частные методики в рамках любого из учебных предметов, одним из которых является информатика. Именно информатика изучает вопросы информатизации общества, поиск, переработку и преобразование информации во всех ее видах.

Программирование, как раздел информатики, предназначен для создания конкретных алгоритмов решения ее задач. Обучение программированию более всего нуждается в преобразовании в соответствии с условия-

ми, необходимыми для успешной реализации процесса информатизации на всех образовательных уровнях.

В то же время достижению данной цели препятствует целый ряд *общих противоречий*:

- между требованием к высокому научному уровню учебных знаний с обязательным выделением взаимосвязей между отдельными объектами и низким уровнем востребованности полученных знаний при практической их реализации;

- между необходимостью формирования у обучаемых умений осуществления проектной деятельности (в соответствии с требованиями информационного общества) и отсутствием соответствующих методик, позволяющих развивать таких умения, сохраняя высокий уровень фундаментальности знаний;

- между необходимостью развития мотивационной сферы обучаемых и недостаточно разработанными теорией и методикой ее формирования.

Разработка методики обучения программированию, направленной на развитие самостоятельности, критического мышления, творческой активности – это сложный процесс соотнесения синтаксиса и семантики языков программирования, принципов разработки программ, специализированных алгоритмов с методами их усвоения. В настоящее время для актуализации и закрепления знаний, умений и навыков по программированию используются небольшие задачи (решение каждой из них не превышает одного занятия). Однако в формировании мотивационной сферы обучаемых при этом возникают существенные затруднения, не создаются условия для самостоятельного приобретения знаний учащимися, объем полученных знаний не находит своего применения в конкретных жизненных ситуациях.

Настоящее исследование посвящено проблемам теории и методики обучения программированию школьников старших классов, в частности,

разработке методики, позволяющей оптимально сочетать изучение фундаментальных знаний с формированием практических умений и навыков.

Основу данной методики составляет "метод проектов", классическое понимание которого представляет собой широкое внедрение учебных проектов (проблемных задач высокой сложности, рассчитанных на долговременность решения) в процесс обучения. Он ориентирован на гуманистичный, индивидуальный подход к личности ученика, основанный на идеях Ж.Ж. Руссо, Дж. Дьюи, У. Килпатрика, предлагающих индуктивный путь учебного познания (учение через открытие, через разрешение проблемных ситуаций). Применение метода проектов характеризуется высоким мотивационным потенциалом, возможностями развития творческих способностей и самостоятельности, что позволяет рассматривать его как эффективный способ познания. В обучении программированию применение этого метода приобретает особую важность. Решение глобальных задач информатизации предполагает создание столь же объемных программных проектов, обладающих высоким уровнем системности и разрабатываемых группами программистов.

Однако внедрение метода проектов в процесс обучения программированию учащихся старших классов сопровождается рядом частных противоречий, требующих своего разрешения. К ним относятся противоречия:

- между востребованностью метода проектов и слабой разработанностью его процессуально-результативной технологической базы, что затрудняет применимость данного метода как педагогической технологии, гарантирующей достижение запланированных результатов обучения и воспитания;

- между требованием к систематичности и фундаментальности знаний в старших классах общеобразовательной школы и усложнением организации получения учеником таких знаний при обучении на основе метода проектов по сравнению с традиционными методиками;

- между широкими возможностями метода проектов в развитии мотивационной сферы обучаемых и недостаточной разработанностью теоретических положений, обосновывающих направленность познавательной активности школьника на достижение целей учебного познания через реализацию программного проекта.

Каждое из противоречий является фактором, сдерживающим применение метода проектов. Во-первых, слабая разработанность технологической базы затрудняет обучение школьников проектной деятельности на основе системного подхода. Во-вторых, фундаментальность и системность научных знаний является обязательным условием научного подхода к проектированию больших систем, без чего невозможно создание объемных, сложных и высококачественных программных продуктов. В третьих, слабый контроль над направленностью познавательной активности школьника в старших классах снижает ее эффект с точки зрения получения обучаемым запланированных в программе курса знаний, умений и навыков.

Таким образом, актуальность исследования обусловлена социальным заказом общества на формирование личности с высоким уровнем научности приобретенных в процессе обучения знаний, наличием умений и навыков применения системного подхода к решению актуальных проблем, творческой активностью и самостоятельностью в проведении анализа и синтеза накопленной информации.

Цель исследования - разработка и обоснование методики применения метода проектов в обучении школьников старших классов общеобразовательных школ программированию в курсе информатики.

Объект исследования – процесс обучения школьников старших классов информатике.

Предмет исследования – применение метода проектов обучении программированию учащихся старшего звена.

Гипотеза исследования: Если методика обучения программированию в курсе информатики будет основана на применении метода проектов, а организация познавательной деятельности школьников будет соответствовать требованиям технологичности, то это обеспечит возрастание познавательных потребностей учащихся и повысит эффективность процесса обучения.

Критерии эффективности предлагаемой методики:

1. Возрастание познавательных потребностей учащихся определялось по следующим показателям: осознание необходимости включения в учебно-познавательную деятельность, готовность к учебно-познавательной деятельности, проявление инициативы, удовлетворенность процессом учебного труда и его результатами.

2. Эффективность процесса обучения программированию устанавливалась путем комплексного анализа изменения уровня знаний, умений и навыков в области программирования.

Задачи исследования:

1. Провести сравнительный анализ существующих методик и определить их основные характеристики в сравнении с методом проектов в процессе обучения программированию.

2. Проанализировать возможность применения метода проектов как педагогической технологии, разработать ее технологическую карту и технологическую схему.

3. Выявить условия эффективного использования метода проектов в процессе обучения программированию в курсе информатики.

4. Разработать методику обучения программированию в курсе информатики на основе метода проектов.

5. Провести педагогический эксперимент в целях проверки эффективности предложенной методики и основных положений диссертации.

Методологическую основу исследования составляют теория и методика обучения информатике (А.П. Ершов, А.А. Кузнецов, И.Е. Подчиненов, В.Ф. Шолохович, А.Г. Гейн, А.Г. Кушнеренко, В.К. Белошапка, С.А. Бешенков, Т.А. Бороненко, А.И. Бочкин, В.Г. Коуров, А.С. Лесневский, В.М. Монахов и др.); исследования в области педагогических технологий (В.П. Беспалько, И.П. Волков, В.М. Монахов, Г.К. Селевко, Е.С. Полат, В.В. Гузеев и др.) и новых информационных технологий в образовании (В.Г. Житомирский, Я.А. Ваграменко, И.М. Бобко, Е.З. Власова, В.А. Горбаренко, И.Б. Готская, И.В. Гребнев, С.А. Жданов, В.А. Извозчиков, А.А. Кузнецов, Э.И. Кузнецов, М.И. Лапчик, И.А. Румянцев, О.Г. Смолянинова, С.А. Феофанов и др.); теоретические основы информатики (Ф.Л. Бауэр, Г. Гооз, М. Брай, В.М. Глушков и др.); теория и практика программирования (Д. Кнут, Н. Вирт, Б. Керниган, Д. Райли, М. Канту, Т. Гуч, Н. Рубенкинг и др.).

Методы исследования:

- изучение и анализ психолого-педагогической, научной, методической литературы, материалов Internet по рассматриваемой проблеме;
- анализ содержания программ, учебных планов, структуры урока информатики в средней школе и в педагогическом вузе;
- обобщение опыта инновационной деятельности учителей и преподавателей;
- наблюдение, беседа, анкетирование обучаемых, учителей и преподавателей с целью выяснение целесообразности использования метода проектов в обучении;
- проведение педагогического эксперимента с использованием методов математической статистики.

Достоверность и обоснованность результатов обеспечиваются:

- использованием предшествующих и научно обоснованных результатов психолого-педагогических и методических исследований;

- выбором взаимодополняющих методов педагогического исследования, соответствующих поставленным задачам;
- опорой на современное содержание информатики, методических исследований по данной теме и психолого-педагогических исследований процесса обучения информатике в школе;
- применением статистических методов обработки данных педагогического эксперимента.

Исследование проводилось в три этапа с 1995 по 2000 год.

На *первом этапе* (1995 – 1996 учебный год) основной целью являлось определение состояния проблемы обучения программированию в старших классах общеобразовательных школ. Констатирующий эксперимент проводился в средних школах Шадринска и Шадринского района, а также в Шадринском государственном педагогическом институте.

На *втором этапе* (1996–1997 учебный год) была разработана методика обучения программированию на основе использования метода проектов. Была разработана технологическая карта процесса обучения, создана методика подготовки спецификаций проектов. Поисковый эксперимент проводился совместно с учителями информатики школ № 4 и № 1 Шадринска и преподавателями ШГПИ.

На *третьем этапе* (1997–2000 учебные годы) был проведен системный формирующий эксперимент, в ходе которого определялась эффективность методики обучения программированию, основанной на использовании метода проектов. Контрольный эксперимент проводился в школах Шадринска, Шадринского района и ШГПИ.

Научная новизна заключается в том, что в отличие от ранее проведенных исследований разработана и обоснована методика обучения программированию в курсе информатики на основе использования метода проектов, эффективность которой обуславливается технологичностью учебного процесса и дифференцированным подходом к его организации.

Теоретическая значимость:

1. Выделены классификационные признаки метода проектов в обучении программированию, разработаны технологическая карта и технологическая схема процесса обучения программированию на основе использования метода проектов.

2. Выделены типологические признаки учебных программных проектов (цели создания, доминирующие методы, характер творческой деятельности, координации проекта и т.д.).

3. Определены требования к построению спецификаций проектов в содержательном (проблема, актуальность, цели, задачи, тип и т.д.) и в процессуальном плане (структура деятельности, средства, методы и т.д.).

4. Выделены уровни готовности школьников к осуществлению проектной деятельности и определены их критерии.

Практическая значимость:

1. Разработаны методические рекомендации по реализации структуры технологической схемы процесса обучения, построенного на основе метода проектов.

2. Разработаны рекомендации по планированию уроков, разработке тематики проектов в соответствии с возрастными особенностями и возможностями обучаемых, а также функциональными особенностями изучаемых языков программирования.

3. Предложена методическая разработка курсов программирования и методики обучения информатике для подготовки учителей к осуществлению обучения на основе использования метода проектов.

Апробация основных выводов исследования осуществлена обсуждением их на всероссийской научно-практической конференции "Информатизация образования 99" (г. Шадринск, 22-24 марта 1999), всероссийской научно-практической конференции "Информатизация образования 2001" (г. Екатеринбург, 13-16 февраля 2001), всероссийской научно-

практической конференции "Повышение эффективности подготовки учителей физики, информатики, технологии в условиях новой образовательной парадигмы" (г. Екатеринбург, 21-22 апреля 2001), научно-методических семинарах аспирантов и преподавателей Шадринского государственного института и Уральского государственного педагогического университета (1995-2001 гг.), заседаниях методических объединений учителей Шадринска и Шадринского района (1995-2001гг).

На защиту выносятся следующие положения:

1. Применение метода проектов при обучении программированию в курсе информатики средней школы активизирует учебную деятельность учащихся и повышает их познавательные потребности.

2. Методика обучения программированию, основанная применении метода проектов, повышает эффективность учебного процесса, если организация познавательной деятельности обучаемых отвечает требованиям технологичности.

ГЛАВА 1. ИСПОЛЬЗОВАНИЕ МЕТОДА ПРОЕКТОВ В УЧЕБНОМ ПРОЦЕССЕ

Первая глава посвящена анализу особенностей метода проектов и его дидактическим возможностям в учебном процессе.

Определяется, что применение метода проектов в обучении старшеклассников программированию в курсе информатики может быть отнесено к педагогическим технологиям при выполнении следующих условий: соответствия требованиям технологичности, выделения классификационных признаков технологии обучения, определения основополагающих принципов построения процесса обучения на основе метода проектов. Строится технологическая карта и схема процесса обучения с использованием метода проектов. Раскрывается типология проектов, вводятся типологические признаки, охватывающие весь спектр учебных проектов.

1.1. Метод проектов как педагогическая технология

Для реализации новой парадигмы образования, в соответствии с которой применяются принципы индивидуализации и дифференциации обучения, обеспечивается вариативность и альтернативность образовательных систем, гибкость и динамичность программной документации, ее прогностичность и адаптированность к имеющимся условиям, индивидуальным интересам и способностям обучающихся, требуется поднять как общий, так и частнопредметный уровень образования. Особую важность в данном процессе представляет собой информатика, как одна из самых молодых и перспективных наук современности.

Новый образовательный уровень должен соответствовать социально-экономическим потребностям общества, что подразумевает построение процесса обучения на основе максимальной активизации форм познава-

тельной деятельности, развития познавательных потребностей, интересов, умения творчески применять усвоенные знания, умения и навыки в своей практической деятельности.

В связи с этим, одной из важнейших задач педагогической науки является разработка эффективных методов обучения, которые должны быть, с одной стороны, высокотехнологичными, а с другой - обладать гибкостью в возможности корректировки и трансформации к конкретным условиям. На сегодняшний день такие условия обеспечивают только новые информационные педагогические технологии.

Понятия "педагогическая технология" и "технология обучения" (от английского "an educational technology" – "образовательная технология") обладают в настоящее время высоким уровнем синонимичности. Это позволяет нам опираться на понятие "педагогическая технология" во всех случаях, когда исследователи употребляют в своих работах термины "технология обучения" или "образовательная технология":

Первые упоминания о педагогических технологиях исследователи относят к 50-м годам 20-го столетия. Первоначально данное понятие связывались с технизацией процесса обучения, активным внедрением ТСО и программированного обучения. Впоследствии понятие "педагогическое технология" было в значительной степени расширено. Рассмотрим его основные определения.

"Технология – это совокупность приемов, применяемых в каком-либо деле, мастерстве, искусстве" [122, с. 346].

"Педагогическая технология – совокупность психолого-педагогических установок, определяющих социальный набор и компоновку форм, методов, способов и приемов обучения, воспитательных средств; она есть организационно-методический инструментарий педагогического процесса" [105, с.142].

"Педагогическая технология – это содержательная техника реализации учебного процесса" [18, с. 45].

"Педагогическая технология – это продуманная во всех деталях модель совместной педагогической деятельности по проектированию, организации и проведению учебного процесса с безусловным обеспечением комфортных условий для учащихся и учителя" [114, с.4].

"Педагогическая технология – это последовательная взаимосвязанная система действий педагога, направленных на решение педагогических задач, или как планомерное и последовательное воплощение на практике заранее спроектированного педагогического процесса" [129, с. 330].

По определению ЮНЕСКО, "педагогическая технология - это системный метод создания, применения и определения всего процесса преподавания и усвоения знаний с учетом технических и человеческих ресурсов, их взаимодействие, ставящее своей задачей оптимизацию форм образования".

"Технология обучения представляет собой систему материальных и идеальных (знания) средств, используемых в обучении, и способы функционирования этой системы" [109].

"Под технологией обучения понимается законосообразная педагогическая деятельность, реализующая научно- обоснованный проект дидактического процесса и обладающая более высокой степенью эффективности, надежности и гарантированности результата по сравнению с традиционными методиками обучения" (Фатхи Т.Б. <http://www.mis.rsu.ru/conf/1999a/1-19.htm>)

Таким образом, концептуальные подходы к определению понятия "педагогическая технология" различны, что поставило перед нами задачу составить рабочее определение данного понятия. Для этого мы воспользовались известной в исследованиях методикой контент-анализа, суть которой состоит в выделении основных категорий (признаков) рассматриваем-

мого понятия у авторов и количественный подсчет наиболее значимых признаков, встречающихся в текстах различных определений (Таблица 1)

Таблица 1

Автор	Признак понятия					
	Системность	Гуманистичность	Процессуальность	Используемые ресурсы	Массовость и гарантированность результатов	Уровень субъект-субъектных отношений
Ожегов			+			
Лихачев Б.Т.			+		+	
Беспалько В.П.	+		+		+	
Монахов В.М.	+	+	+		+	+
Сластенин	+		+		+	
ЮНЕСКО	+		+	+	+	+
Машбиц	+		+	+		
Фатхи			+		+	
Итого:	5	1	8	2	6	2

Результаты проведенного контент-анализа показали, что наиболее значимыми признаками в определениях педагогической технологии авторы считают – **процессуальность** (педагогическая *деятельность*, или *метод обучения*), **массовость и гарантированность результатов**, то есть возможность применения данного метода в различных внешних условиях с гарантией достижения запланированного результата, и **системность**, что определяет данный метод как структурно организованный комплекс, компоненты которого находятся в сложных взаимоотношениях друг с другом и взаимодействуют с внешней средой как единый объект.

Приведем наиболее существенные признаки педагогических технологий:

1) технология подразумевает организацию учебного процесса, основанную на взаимосвязанной деятельности учителя и ученика с учетом принципов индивидуализации и дифференциации, оптимальной реализации человеческих и технических возможностей, диалогического общения;

2) элементы педагогической технологии должны быть воспроизводимы и гарантировать достижение планируемых результатов (государственного стандарта) всеми обучаемыми;

3) органической частью педагогической технологии являются диагностические процедуры, содержащие критерии, показатели и инструментальный измерения результатов деятельности [129, с. 331].

На основе контент-анализа мы составили определение, которого будем придерживаться в дальнейшем при обсуждении результатов нашего исследования:

Педагогическая технология – это системный метод организации учебного процесса, обеспечивающий гарантированное достижение запланированных результатов.

Любая педагогическая технология должна удовлетворять некоторым основным методологическим требованиям (критериям технологичности).

Концептуальность. Каждой педагогической технологии должна быть присуща опора на определенную научную концепцию, включающую философское, психологическое, дидактическое и социально-педагогическое обоснование достижения образовательных целей.

Системность. Педагогическая технология должна обладать всеми признаками системы: логикой процесса, взаимосвязью всех его частей, целостностью.

Управляемость предполагает возможность диагностического целеполагания, планирования, проектирования процесса обучения, поэтапной диагностики, варьирования средствами и методами с целью коррекции результатов.

Эффективность. Современные педагогические технологии существуют в конкурентных условиях и должны быть эффективными по результатам и оптимальными по затратам, гарантировать достижение определенного стандарта обучения.

Воспроизводимость подразумевает возможность применения (повторения, воспроизведения) педагогической технологии в других однотипных образовательных учреждениях, другими субъектами [146, с. 31].

Процесс описания и анализа педагогической технологии производится на основе следующей структуры:

1. **Название технологии**, отражающее основные качества, принципиальную идею, существо применяемой системы обучения, наконец, основное направление модернизации учебно-воспитательного процесса.

2. **Идентификация** данной педагогической технологии в соответствии с принятой систематизацией (классификационной схемой [146, с. 25]).

3. **Концептуальная часть** (краткое описание руководящих идей, гипотез, принципов технологии, способствующее пониманию, трактовке ее построения и функционирования):

- ✓ целевые установки и ориентации;

- ✓ основные идеи и принципы (основной используемый фактор развития, научная концепция усвоения);

- ✓ позиция ребенка в образовательном процессе.

4. **Особенности содержания образования:**

- ✓ ориентация на личностные структуры (знания, умения и навыки; способы умственных действий; самоуправляющиеся механизмы личности; сфера эстетических и нравственных качеств личности; действенно-практическая сфера личности);

- ✓ объем и характер содержания образования;

- ✓ дидактическая структура учебного плана, материала, программ, формы изложения.

5. Процессуальная характеристика:

- ✓ особенности методики, применения методов и средств обучения;
- ✓ мотивационная характеристика;
- ✓ организационные формы образовательного процесса;
- ✓ управление образовательным процессом (диагностика, планирование, регламент, коррекция);
- ✓ категория учащихся, на которых рассчитана технология.

6. Программно-методическое обеспечение:

- ✓ учебные планы и программы;
- ✓ учебные и методические пособия;
- ✓ дидактические материалы;
- ✓ наглядные и технические средства обучения;
- ✓ диагностический инструментарий [146, с. 32].

Исследования в области педагогических технологий наглядно показывают, что структура методики, предназначенной для массового применения в процессе обучения должна удовлетворять вышеизложенным принципам. Таким образом, одним из важнейших критериев эффективности любой методики обучения является соответствие ее требованиям, предъявляемым к педагогической технологии.

1. НАЗВАНИЕ ТЕХНОЛОГИИ

Наше исследование касается возможностей применения метода проектов в обучении программированию. Остановимся на генезисе понятия "метод проектов", его составе и структуре.

В начале 20-го века по инициативе Е. Паркхерст появилась педагогическая система, известная в настоящее время как "Дальтон-план" (от названия американского города Долтон в штате Массачусетс). По своей педагогической направленности дальтон-план, как и некоторые другие альтернативные школы, противостоит традиционной линии Гербарта, в кото-

рой декларируется ведущая роль педагога и систематичное изучении наук дедуктивными методами. Основанный на идеях Ж.-Ж. Руссо, Дж. Дьюи, У. Килпатрика, дальтон-план предусматривает индуктивный путь учебного познания, учение через открытие, через разрешение проблемных ситуаций. Близкий по основным концепциям к школе Френе, дальтон-план не обладает столь жестким культом труда, гораздо большее внимание уделяется ответственности ученика за решение выбранной им самим задачи.

По мнению Лилиан Леферинк (нидерландская ассоциация дальтоновских школ) [<http://www.emissia.spb.su/offline/a121.htm>] "образовательные задачи дальтон-плана, в первую очередь, вытекают из целей воспитания, и помимо этого - из особенностей детского развития, взглядов на учебные предметы, теорий обучения, жизненного мира учащихся, а также их потребностей. Дальтон-план стремится предоставить каждому ребенку возможности развития в моторной, эмоциональной, социальной, когнитивной и нравственной сферах. ... дети работают над достижением целей обучения с помощью самостоятельного решения проблем, а также применяя свои знания и навыки. Они учатся пользоваться поисковыми системами для получения информации и приобретают умения задействовать готовые знания и навыки."

В СССР дальтон-план получил свое дальнейшее развитие в 20-х годах и стал известен как "Метод проектов". Воплотив в себе идеи дальтон-плана, школы Френе и технологии мастерских, метод проектов стал одной из наиболее часто используемых педагогических инноваций в советской России. К сожалению, непродуманность внедрения, отсутствие четкой технологической базы метода проектов, а также чрезмерная идеологизация в значительной степени нивелировала эффективность данного метода. По мнению Н.В. Матяш [108] при внедрении метода проектов были допущены следующие ошибки: 1) метод проектов рассматривался как единственное средство обучения, 2) отсутствовали квалифицированные педагогические

кадры, способные применять метод проектов в обучении. В 1931 году метод проектов был осужден и запрещен к применению. С этого момента основополагающей концепцией школьного образования стала линия Гербарта.

В настоящее время дальтон-план, школа Френе, Йена-план школа, школа Монтессори и Вальфдорфская школа являются пятью наиболее распространенными инновационными (альтернативными) направлениями в обучении в США и странах Европы.

Метод проектов в том или ином виде используется в России, имеются, хотя и в небольшом количестве, дальтон-школы. Большое внимание уделяется методу проектов в системе технологического образования. Количество публикаций в сети Internet по методу проектов на русском языке исчисляется тысячами, а по дальтон-плану (на английском и русском языках) – сотнями тысяч.

В то же время технологические аспекты метода проектов остаются неразработанными. В работе Г.К. Селевко "Современные образовательные технологии" 1998 года выпуска отсутствуют упоминания о дальтон-плане, а метод проектов рассматривается только как частная особенность некоторых педагогических школ. Возникающее противоречие между мировым опытом, востребованностью метода проектов, практическими разработками российских педагогов и отсутствием технологической базы данного метода, требует разрешения.

Основным недостатком дальтон-плана и метода проектов, как и других альтернативных школ, основанных на обучении индуктивными методами, является сложность в организации получения учеником систематичных, фундаментальных знаний. Данная система обучения характеризуется высоким уровнем полученных умений и навыков, и фрагментарным, хотя и глубоким уровнем усвоенных знаний. Именно поэтому дальтон-план в основном используется в начальной школе и на первой ступени обучения.

К положительным результатам относится высокий уровень мотивации, познавательной активности школьников, развития творческих способностей, самостоятельности. В обучении по линии Гербарта ситуация прямо противоположна. Академичность получаемых знаний, их отрыв от реальной жизни понижает активность обучаемых, желание учиться, узнавать новое.

По нашему мнению, разрешением данного противоречия будет интеграция метода проектов с линией Гербарта, формулировка понятий "учебный проект" и "метод проектов" в соответствии с предлагаемой концепцией, анализ применения метода проектов как педагогической технологии и разработка методики его использования в обучении программированию в соответствии с требованиями к педагогической технологии. Это позволит ввести метод проектов в структуру учебной деятельности школьников старших классов и студентов педагогических ВУЗов.

Сформулируем определения таких понятий, как "учебный проект" и "метод проектов".

Учебным проектом будем называть проблемную задачу, процесс анализа и решения которой может быть алгоритмизован, обладает свойствами поэтапности и длительности, опирается на систему научных знаний в изучаемой области. Алгоритмизованность разработки проекта подразумевает организацию плана исследовательской деятельности по решению задачи, разбиение ее на подзадачи, многие из которых обладают всеми свойствами исходного проекта и решаются методами, применяемыми ко всему проекту. Под поэтапностью будем понимать спиралевидно-циклический характер создания проекта. При этом результат осуществления плана деятельности может вступать в противоречие со всем планом или отдельными его пунктами. Следствием данного процесса является переработка исходного плана и его повторная реализация уже на более высоком уровне. Длительность подразумевает большую продолжительность создания учебного проекта (от 2-х занятий до года), является следствием

свойств алгоритмизованности и поэтапности проекта. Данное свойство служит одним из средств повышения эффективности метода проектов с точки зрения гештальтпсихологии (эффект "прерванного действия"). Опора процесса разработки проекта на систему научных знаний является необходимым условием успешности обучения в старших классах общеобразовательной школы. Данный аспект на сегодняшний день рассмотрен слабо и нуждается в дополнительных педагогических исследованиях.

В технологическом образовании, по мнению Н.В. Матяш, метод проектов – " это система обучения, гибкая модель организации учебного процесса, ориентированная на самореализацию личности учащегося путем развития его интеллектуальных и физических возможностей, волевых качеств и творческих способностей в процессе создания под контролем учителя новых товаров и услуг, обладающих субъективной или объективной новизной и имеющих практическую значимость" [107, с.38]. Там же под методом творческих проектов понимается "... деятельность учащегося, в процессе которой под руководством учителя создаются материальные или духовные ценности, обладающие субъективной, а возможно, и объективной новизной ..."

Е.С. Полат, говоря о методе проектов, определяет его как "... способ достижения дидактической цели через детальную разработку проблемы (технология), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом" [133].

На основе вышеизложенного сформулируем определение метода проектов:

Метод проектов – это педагогическая технология, основанная на широком внедрении в процесс обучения учебных проектов, обладающих социальной и личностной значимостью.

2. КЛАССИФИКАЦИОННЫЕ ПАРАМЕТРЫ ТЕХНОЛОГИИ, ОСНОВАННОЙ НА МЕТОДЕ ПРОЕКТОВ

По уровню применимости: частнометодическая.

В общем случае можно сказать, что данная технология является общеметодической, но в нашем исследовании мы будем ее рассматривать применительно к процессу обучения программированию, с точки зрения частной методики.

По философской основе: прагматическая.

В целях данной технологии четко не заложены философские концепции.

Концепция прагматизма исходит из того, что интеллектуальные и нравственные качества личности заложены в ее уникальной природе и их проявление связано прежде всего с индивидуальным опытом человека. Отсюда задача образования – в количественном росте данных ей от природы способностей, качеств и ее индивидуального опыта как главного условия самореализации (Д. Дьюи).

По ведущему фактору психического развития: биогенная + социогенная + психогенная.

Биогенная технология предполагает, что развитие психики определяется генетическим кодом; внешняя среда лишь реализует наследственные данные.

Социогенная представляет личность как "tabula rasa", на которой записывается социальный опыт человека, результаты обучения.

Психогенная – это технология результат развития в которых определяется, главным образом, самим человеком, его предшествующим опытом, психологическими процессами самосовершенствования.

По концепции усвоения: ассоциативно-рефлекторная + интериоризационная.

Согласно ассоциативно-рефлексивной теории, усвоение знаний, формирование навыков и умений, развитие личностных качеств человека есть процесс образования ассоциаций. В рамках этой теории разработаны теории формирования понятий, деятельностная теория учения.

По ориентации на личностные структуры: информационная + операционная + эвристическая + прикладная + технология саморазвития.

Информационная технология ориентирована на формирование научных знаний, умений и навыков по программированию.

Операционная – на формирование способов умственных действий.

Эвристическая – на развитие творческих способностей.

Прикладная – на формирование действенно-практической сферы.

Технология саморазвития – на формирование самоуправляющихся механизмов личности.

По характеру содержания и структуры: обучающая, светская, общеобразовательная.

Обучающая технология направлена на формирование знаний, умений и навыков по некоторой научной дисциплине.

Термин "светская" означает отсутствие религиозной основы у данной технологии.

Общеобразовательная технология может быть использована при обучении в средней общеобразовательной школе, а также в современных гимназиях, лицеях, специализированных школах.

По организационным формам: классно-урочная, групповая, индивидуальная.

Организация обучения происходит по классно-урочной системе.

Педагогическая технология с групповой или индивидуальной формой организации обучения предполагает наличие групповых или индивидуальных методов работы.

По типу управления познавательной деятельностью: "программное обучение" (циклическое, направленное, автоматизированное).

Циклическое управление – это управление с контролем, самоконтролем и взаимоконтролем.

Направленное – с индивидуальным управлением.

Автоматизированное – с помощью учебных средств.

По подходу к ребенку: личностно-ориентированная.

Данная технология ставит в центр школьной образовательной системы личность ребенка, обеспечение комфортных, бесконфликтных и безопасных условий ее развития, реализации ее природных потенциалов.

По преобладающему методу: проблемная, саморазвивающаяся, творческая.

Проблемная технология предполагает использование в процессе обучения проблемных методов.

Саморазвивающаяся технология означает не жесткий характер связи между отдельными ее этапами с возможностью их изменения в соответствии с поставленной задачей.

Творческая технология основывается на творческом подходе не только со стороны обучаемых при обучении, но и от учителя в процессе подготовки.

По категории обучающихся: массовая + продвинутого уровня.

Массовая рассчитана на усредненного ученика.

Продвинутого уровня – на школьников, занимающихся углубленным изучением информатики (программирования).

3. ЦЕЛЕВЫЕ ОРИЕНТАЦИИ

- ✓ Обеспечение активности учебного процесса.
- ✓ Достижение высокого уровня усвоения содержания.
- ✓ Формирование знаний, умений и навыков.

- ✓ Адаптация обучения к особенностям различных групп учащихся.
- ✓ Обучение каждого на уровне его возможностей.
- ✓ Формирование исследовательских умений, умений принимать оптимальные решения.
- ✓ Развитие познавательных и творческих способностей.
- ✓ Усвоение способов самостоятельной деятельности.
- ✓ Развитие коммуникативных качеств личности.

4. КОНЦЕПТУАЛЬНЫЕ ПОЛОЖЕНИЯ

- ✓ Учащиеся усваивают материал как результат удовлетворения возникающей у них потребности в знаниях, являясь активным субъектом своего обучения.
- ✓ Условиями успешности обучения являются:
 - проблемизация учебного материала;
 - активность обучаемых;
 - связь обучения с жизнью, будущей профессиональной деятельностью.

5. ОСОБЕННОСТИ СОДЕРЖАНИЯ

Обучение программированию с использованием метода проектов основано на создании больших программных комплексов, разработка которых предполагает активное применение основных знаний, умений и навыков, приобретаемых обучаемым за данный промежуток учебного времени. Этот аспект накладывает на содержание образования некоторые характерные особенности.

Одной из таких особенностей является наличие двух уровней владения знаниями, умениями и навыками: высокий и массовый. Высокий подразумевает больший (по сравнению со стандартом) объем учебного материала, умение решать нестандартные задачи, владение навыками програм-

мирования сложных систем, не предусмотренных программой курса. Массовый же подразумевает уровень знаний, умений и навыков, предусматриваемый государственным стандартом для данного учебного заведения.

Другая особенность заключается в том, что логика изложения учебного материала, его структурные характеристики строятся на основе процесса создания проекта. С другой стороны, выбор программного проекта и разработка его спецификации основаны на требованиях учебной программы, предусматривающей усвоение определенного набора знаний, умений и навыков. Диалектичность данного процесса во многом обуславливает постоянное развитие познавательной деятельности обучаемых.

6. ОСОБЕННОСТИ ТЕХНОЛОГИИ

Метод проектов является логическим развитием проблемных методов обучения.

Основной особенностью этой технологии является тот факт, что ученики изначально настроены на достижение некоторой цели, преодоление противоречий, обусловленных наличием проблемной ситуации, которая может быть создана любым субъектом (субъектами) учебной деятельности. Роль учителя заключается, с одной стороны, в регуляции и контроле данного процесса, с другой – в передаче обучаемым знаний, развитии учебных умений и навыков.

Таким образом, применение метода проектов в обучении программированию будем рассматривать как педагогическую технологию, в следствии чего становится возможным:

1. Произвести массовое внедрение метода проектов в процесс обучения.

2. Придерживаясь фундаментальной, академичной, традиционной системы учебных знаний, избегать их отрыва от реальной деятельности, добиваясь тем самым глубокого и надежного усвоения.

3. Развивать познавательную и творческую активность обучаемых, умения самостоятельно мыслить, планировать структуру деятельности.

4. Развивать коммуникативные способности, умение работать в группе.

5. Вести профессиональную разработку прикладного программного обеспечения для удовлетворения личных потребностей, а также нужд учебного заведения или его отдельных структурных единиц.

1.2. Типология проектов, используемых в образовательном процессе

Для дальнейшего анализа метода проектов как средства организации образовательного процесса необходимо осуществить классификацию учебных проектов. Выделение с помощью характеристических признаков типов учебных проектов, позволит построить расширенный план деятельности (спецификацию) по реализации проекта. Методы построения спецификаций проекта с учетом его типа, цели создания, возраста обучаемых и т.д. будут рассмотрены во второй главе диссертации.

Определим основные типологические признаки проектов, используемых для обучения программированию.

1. Цели создания проекта.
2. Методы, доминирующие в процессе разработки проекта.
3. Характер творческой деятельности при реализации проекта.
4. Характер координации проекта.
5. Характер контактов.
6. Количество участников проекта.
7. Продолжительность разработки проекта.

1. ЦЕЛИ СОЗДАНИЯ ПРОЕКТА.

В соответствии с первым типологическим признаком – **цель создания проекта** – выделим учебные и прикладные типы проектов (Рис. 1).

Математически это отношение можно представить следующим образом: Прикладные \subset Учебные.

Учебные.

Все проекты, созданные в учебных целях являются учебными. В своем исследовании мы имеем дело только с учебными проектами.



Рис. 1. Взаимоотношение проектов, выделенных по классификационному признаку "цель создания"

Примеры учебных проектов, не обладающих прикладной значимостью:

1. Логическая игра "Балда". Язык программирования Turbo Pascal. Автор: Попов Е.

2. Текстовый редактор. Язык программирования – Turbo Pascal. Авторский коллектив: Тенигин А., Баландин А.

Прикладные.

Эти проекты отличает четко обозначенный с самого начала результат деятельности учащихся, ориентированный на их личные или социальные интересы.

Такой проект требует хорошо продуманной структуры, даже сценария всей деятельности его участников с определением функций каждого из них, четких выводов и участия каждого в оформлении конечного продукта. Здесь особенно важна хорошая организация координационной работы в плане поэтапных обсуждений, корректировки совместных и индивидуальных усилий, в организации презентации полученных результатов и возможных способов их внедрения в практику, организация систематической внешней оценки проекта.

Примеры прикладных проектов:

1. Интерпретаторы нормальных алгоритмов Маркова (НАМ) и машины Тьюринга. Язык программирования – Delphi. Автор: Барыкин В.
2. Программа "Деканат". язык программирования – FoxPro. Автор: Балахнов М.

2. МЕТОДЫ, ДОМИНИРУЮЩИЕ В ПРОЦЕССЕ РАЗРАБОТКИ ПРОЕКТА.

Рассмотрим второй комплекс типологических признаков. В соответствии с ним выделим следующие типы проекта: исследовательский, творческий, информационный, творческо-информационный, творческо-исследовательский, информационно-исследовательский и творческо-информационно-исследовательский. Необходимо отметить, что в данном случае речь идет именно о доминировании метода деятельности, используемого при реализации проекта над всеми остальными. Это означает, что при создании любого проекта происходит процесс исследования в некоторой области знаний, поиск информации, творческая переработка полученных материалов или творческий подход к их оформлению. Таким образом каждый из выделенных процессов: исследование, творчество и информационный поиск присутствует при реализации проекта в той или иной степени. Наша задача заключается в выделении свойств проектов, для которых тот или иной процесс является доминирующим и в описании его основных характеристических признаков.

Исследовательский проект.

Цель этого проекта – изучение конкретной части теоретического материала и представление результатов в виде программы (комплекса программ), реализующей отдельные компоненты теории. Исследовательские проекты наиболее эффективно можно использовать при работе с группой обучаемых, углубленно изучающих предмет.

Этот тип проекта требует от спецификации хорошо продуманной структуры, актуальности предмета исследования для всех участников, социальной значимости, продуманных методов, в том числе экспериментальных, методов обработки результатов. Такие проекты полностью подчинены логике исследования и имеют структуру, приближенную или полностью совпадающую с подлинным научным исследованием: аргументация актуальности темы, определение проблемы исследования, его предмета и объекта, обозначение задач исследования в последовательности принятой логики, определение методов исследования, источников информации, определение методологии исследования, выдвижение гипотез решения обозначенной проблемы, определение путей ее решения, в том числе экспериментальных, опытных, обсуждение полученных результатов, выводы, оформление результатов исследования, обозначение новых проблем на дальнейший ход исследования [121].

Примеры исследовательских проектов:

1. Реализация API-функций для работы с регионами в операционной системе MS-DOS. Язык программирования – C++. Автор: Чеботарев С.

2. СУБД "Студенты". Язык программирования – Object Pascal в среде Delphi. Автор: Кириллов А.

Творческий проект.

При его реализации необходимо проявить в полной мере творческие способности. В качестве результата могут быть представлены сайты, текстовые документы, несложные конструктивные программные продукты, требующие живого воображения от создателей в реализации интерфейса или информационном наполнении.

Такой компонент проекта, как правило, не требует от спецификации детально проработанной структуры совместной деятельности участников, она только намечается и далее развивается, подчиняясь жанру конечного результата, обусловленный этим жанром и принятой группой логике со-

вместной деятельности, интересам участников проекта. В данном случае следует договориться о планируемых результатах и форме их представления. Однако оформление результатов проекта требует четко продуманной структуры и в виде документации по программному проекту, методического пособия и т.д.

Примеры творческих проектов:

1. Сайт "Берестяное барокко". Язык разметки гипертекста HTML, языки программирования JavaScript, Java Автор: Махнюк М.

2. "Автоматизированная среда для создания тестов и анкет". Язык программирования – Delphi. Авторский коллектив: Попов Е., Кириллов А.

Информационный проект.

Основная цель данного проекта – сбор, систематизация и обобщение некоторой информации. Результат выполнения проекта должен быть представлен в виде информационной программной системы. Этот тип проектов изначально направлен на сбор информации о каком-то объекте, явлении, на ознакомление участников проекта с этой информацией, ее анализ и обобщение фактов, предназначенных для широкой аудитории. Такие проекты так же, как и исследовательские, требуют хорошо продуманной структуры, возможности систематической коррекции по ходу работы. Они часто интегрируются в исследовательские проекты и становятся их органической частью.

Структура исследовательской деятельности с целью информационного поиска и анализа очень схожа с предметно-исследовательской деятельностью:

- Предмет информационного поиска. Поэтапность поиска с обозначением промежуточных результатов.
- Аналитическая работа над собранными фактами. Выводы. При необходимости - корректировка первоначального направления поиска.
- Дальнейший поиск информации. Анализ новых фактов.

- Обобщение.
- Выводы (и так далее до получения данных, удовлетворяющих всех участников проекта).
- Заключение, оформление результатов (обсуждение, редактирование, презентация, внешняя оценка).

Примеры информационных проектов:

1. "История внедрения новых информационных технологий в г. Шадринске". Форма представления результата: сайт и выступление на фестивале научного творчества студентов, посвященном дню Информациолога. Авторский коллектив: Лобанов Р., Тарасова Я., Третьяков А.

2. Обновление сайта "Новости" Web-сервера кафедры ТМИ Шадринского государственного педагогического института. Форма представления результата: файлы в формате HTML. Авторский коллектив: Махнюк М., Дюков А., Журило Л.

Творческо-информационные проекты.

Цель данного проекта - сбор и анализ информации, творческая ее переработка и преобразование.. Одним из самых интересных примеров таких проектов являются авторские электронные учебники, Internet-газеты, тематические сайты.

Примеры творческо-информационных проектов:

1. Электронный учебник по Window's 95-98. Язык программирования – Object Pascal в среде Delphi. Автор: Спирин И.

2. Шахматный сайт, посвященный популяризации этой игры в городе Шадринске. Язык разметки гипертекста HTML. Автор: Дюков А.

Информационно-исследовательские проекты.

Исследования, заключающиеся в сборе, переработке и представлении информации, организации программных фильтров для доступа к информации, создании сложных программных систем автоматизации доступа к информации лежат в основе проектов данного типа. Отличительной чертой

этих проектов является необходимость анализа не только представляемой в программном продукте информации, но и теоретический анализ методов реализации проекта, обобщение уже накопленного опыта и поиск оптимального решения проблемы.

Примеры информационно-исследовательских проектов:

1. СУБД "Библиотека". Язык программирования – Object Pascal в среде Delphi. Авторский коллектив: Спиридонов Ю., Пантелеев Д.

2. Информационный сайт "Компьютерный ад". Язык разметки гипертекста HTML, языки программирования JavaScript, Java. Автор: Лобанов Р.

Творческо-исследовательские проекты.

Объединяя в себе параметры творческих и исследовательских проектов, данный тип проекта имеет следующий характерный признак – результат, который должен быть получен, на начальных этапах представляется в общих чертах. Обязательным образом указывается только направление исследований, а пути достижения результата (в некоторых случаях – и форма) определяются в процессе творческого поиска.

Примеры творческо-исследовательских проектов:

1. "Talking Cat" – клавиатурный тренажер. Язык программирования – Delphi. Автор: Барыкин В.

2. Логическая игра "Балда". Язык программирования Turbo Pascal. Автор: Попов Е.

Творческо-информационно-исследовательские проекты.

Это самая сложная конфигурация методов деятельности при разработке проекта. Предметом творческого поиска для таких проектов является проблема исследовательского характера, а полученный результат может представлять собой информационную программную систему, предметно-ориентированную среду, инструментальную систему и т.д.

Примеры творческо-информационно-исследовательских проектов:

1. "SAL" – интерпретатор школьного алгоритмического языка, представленный вместе с библиотеками алгоритмов. Язык программирования – Object Pascal в среде Delphi. Авторский коллектив: Попов Е., Хасиев Р.

2. Векторный графический редактор. Язык программирования – Object Pascal в среде Delphi. Автор: Кириллов А.

В соответствии с описанием классификацию проектов по доминирующему методу можно наглядно представить в виде множеств со всеми свойствами их пересечений (Рис. 2)



Рис. 2. Структура классификации проектов по доминирующему методу

3. ХАРАКТЕР ТВОРЧЕСКОЙ ДЕЯТЕЛЬНОСТИ ПРИ РЕАЛИЗАЦИИ ПРОЕКТА.

По характеру творческой деятельности выделим рутинные, квазирутинные и нерутинные типы проектов (Таблица 2).

Примеры рутинных проектов:

1. Создание обучающей программы "Английский без проблем". Язык программирования – Turbo Pascal. Авторский коллектив: Маслов С., Дугин М., Жданова Л., Пирогов Е.

1. Интерпретаторы Нормальных алгоритмов Маркова и машины Тьюринга. Язык программирования Object Pascal в среде Delphi. Автор: Барыкин В.

Примеры квазирутинных проектов:

2. Электронный учебник по Window's 95-98. Язык программирования – Object Pascal в среде Delphi. Автор: Спирин И.

3. Программа для решения задач линейного программирования графическим методом. Язык программирования – Turbo Pascal. Авторский коллектив: Чеботарев С., Сарайкин А.

Примеры нерутинных проектов:

1. "SAL" – интерпретатор школьного алгоритмического языка, представленный вместе с библиотеками. Язык программирования – Object Pascal в среде Delphi. Авторский коллектив: Попов Е., Хасиев Р.

2. Преобразователь БНФ-конструкций в синтаксические диаграммы. Язык программирования – Object Pascal в среде Delphi. Авторский коллектив: Горелов Н., Малышев М.

4. ХАРАКТЕР КООРДИНАЦИИ ПРОЕКТА.

По характеру координации – проекты могут быть двух типов.

С открытой, явной координацией.

В таких проектах координатор проекта участвует в проекте в собственной своей функции, ненавязчиво направляя работу его участников, организуя, в случае необходимости, отдельные этапы проекта, деятельность отдельных его участников (например, если нужно договориться о встрече в каком-то официальном учреждении, провести анкетирование, интервью специалистов, собрать репрезентативные данные и т.д.).

Характер творческой деятельности при реализации проекта

	Рутинные	Квазирутинные	Нерутинные
Определение	<p>Разработчик такого проекта обладает знанием в той или иной форме алгоритма его реализации</p> <p>Модель способа реализации проекта, обеспечивающую решение любой индивидуальной задачи из круга проблем, поставленных в спецификации проекта называют алгоритмом реализации. Все операции в этом алгоритме выполняются всегда вне зависимости от внешних условий</p>	<p>Разработчик такого проекта обладает представлением в той или иной форме квазиалгоритма его реализации</p> <p>Модель способа реализации проекта, с достаточной вероятностью обеспечивающую решение любой индивидуальной задачи из круга проблем, поставленных в спецификации проекта называют квазиалгоритмом реализации проекта. Многие операции в этом квазиалгоритме должны уточняться с помощью творческого поиска</p>	<p>Разработчик такого проекта применяет творческий подход к его реализации.</p> <p>Ни алгоритм, ни квазиалгоритм реализации неизвестен. Необходимо искать нестандартные пути достижения цели</p>
Различия			
Методическая оценка	<p>Эти проекты предназначены для закрепления некоторых знаний, умений и навыков</p>	<p>Наличие квазиалгоритма решения предполагает вероятность того, что проект не будет реализован обычным способом. Такие проекты предназначаются для формирования у обучаемых умений и навыков самостоятельного поиска знаний, развития логического мышления, коммуникативных умений и т.д.</p>	<p>Использование данного типа проекта дает максимальный эффект в развитии когнитивной сферы обучаемых. Однако при его использовании необходима уверенность в наличии потенциальной возможности разработки проекта такого типа обучаемыми, находящимися на конкретном уровне развития и обладающими набором знаний (либо средствами их получения), необходимых для решения проблемы.</p>

Со скрытой координацией (главным образом, телекоммуникационные проекты).

В таких проектах координатор не обнаруживает себя ни в сетях, ни в деятельности групп участников в своей функции. Он выступает как полноправный участник проекта («один из...»).

При обучении программированию в классно-урочной системе образования в подавляющем большинстве случаев программные проекты относятся к открытому типу координации. В случае дистанционного обучения может быть использована скрытая координация.

5. ХАРАКТЕР КОНТАКТОВ.

По характеру контактов проекты разделяются на внутренние (региональные) и международные. В классно-урочной системе образования практически любой учебный программный проект относится к внутреннему типу. При дистанционном обучении программные проекты часто могут быть отнесены к статусу международных.

6. КОЛИЧЕСТВО УЧАСТНИКОВ ПРОЕКТА.

По количеству участников проектов можно выделить три типа проектов: индивидуальные (1 участник), парные (2 участника), групповые (более 2-х участников). Опыт показывает, что наибольший дидактический эффект при создании проектов средней и долгосрочной продолжительности достигается парной разработкой, при которой обучаемые находятся в отношениях "ведущий-ведомый". Разработка проекта группой целесообразна при большом объеме проекта и наличии в группе учащегося, выполняющего функции системного архитектора, либо при псевдогрупповой разработке в классно-урочной системе образования, когда создание одного и того же программного проекта осуществляется учащимися индивидуально (парно) и одновременно.

Примеры индивидуальных проектов:

1. Векторный графический редактор. Язык программирования – Delphi. Автор: Кириллов А.
2. Электронный учебник по Window's 95-98. Язык программирования – Delphi. Автор: Спирин И.

Примеры парных проектов:

1. ППС "Кодирование информации". Язык программирования – Turbo Pascal. Автор: Булычев А., Шалгин А.
2. "Автоматизированная среда для создания тестов и анкет". Язык программирования – Delphi. Авторский коллектив: Попов Е., Кириллов А.

В последнем случае очень важно правильно с методической точки зрения организовать эту групповую деятельность участников проекта (как в локальной группе, так и в объединенной группе участников проекта различных школ, вузов, стран). Роль педагога при этом особенно велика.

Примеры групповых проектов:

1. "Internet-газета факультета информатики". Языки программирования HTML, JavaScript, Java Авторский коллектив: Неустроева Ю., Титова И., Хасиев Р., Кириллов А., Овчиникова О.
2. "История внедрения новых информационных технологий в г. Шадринске". Авторский коллектив: Лобанов Р., Тарасова Я., Третьяков А.

7. ПРОДОЛЖИТЕЛЬНОСТЬ РАЗРАБОТКИ ПРОЕКТА.

По данному признаку проекты делятся на краткосрочные, средней продолжительности и долгосрочные (Таблица 3).

Примеры краткосрочных проектов:

1. Реализация нормальных алгоритмов Маркова с помощью ЭВМ. Язык программирования Turbo Pascal. Авторы: 3 курс факультета информатики. Время исполнения: 2 занятия.

Продолжительность разработки проекта

Краткосрочные (несколько заня- тий)	Средней продол- жительности (от недели до месяца)	Долгосрочные
Для решения не- большой пробле- мы или части бо- лее крупной про- блемы. Такие проекты могут быть разработаны на нескольких занятиях одним учащимся и при- меняются обычно в качестве кон- трольных работ и для закрепления изученного мате- риала	Для решения сред- них по объему и сложности про- блем. Такие проек- ты разрабатывают- ся чаще всего од- ним или парой обучаемых и могут быть использованы при изучении или для закрепления логически завер- шенного блока учебного материа- ла	Такие программные проекты обычно характеризуются либо сложной структурой (внутридис- циплинарные проекты), либо большим объемом (междисципли- нарные проекты), либо обоими свойствами (внутридисциплинар- ные и междисциплинарные про- екты). В обучении программиро- ванию основополагающим пара- метром для таких проектов явля- ется наличие сложной структуры, при разработке и реализации ко- торой актуализируются и приоб- ретаются соответствующие зна- ния, умения и навыки.

Примеры проектов средней продолжительности.

1. «Игра «Балда»». Язык программирования – Turbo Pascal. Авторский коллектив: Тигиняну Е., Федотова Е., Попов Е., Лукин А., Уваров С.
2. «Медицинская карточка». Язык программирования - Object Pascal в среде Delphi. Автор: Горелов Н.

Примеры долгосрочных проектов:

1. "SAL" – интерпретатор школьного алгоритмического языка, представленный вместе с библиотеками. Язык программирования - Object Pascal в среде Delphi. Авторский коллектив: Попов Е., Хасиев Р.
2. Электронный учебник по Window's 95-98. Язык программирования – Object Pascal в среде Delphi. Автор: Спирин И.

Обобщая вышеизложенное представим типологические признаки проектов и соответственные им типы проекта:

1. Признак "цель создания проекта". Типы проектов:

- ✓ учебный $C_{уч}$;
- ✓ учебно-прикладной $C_{уч-прикл}$.

2. Признак "метод, доминирующий в проекте". Типы проектов:

- ✓ исследовательский $M_{иссл}$;
- ✓ творческий $M_{твор}$;
- ✓ информационный $M_{инф}$;
- ✓ творческо-информационный $M_{твор-инф}$;
- ✓ информационно-исследовательский $M_{инф-иссл}$;
- ✓ творческо-исследовательский $M_{твор-иссл}$;
- ✓ творческо-информационно-исследовательский $M_{твор-инф-иссл}$

3. Признак "характер творческой деятельности при реализации проекта.". Типы проектов:

- ✓ рутинные $F_{рут}$
- ✓ квазирутинные $F_{кв_рут}$
- ✓ нерутинные $F_{нерут}$

4. Признак "характер координации проекта". Типы проектов:

- ✓ с открытой, явной координацией $K_{откр}$;
- ✓ со скрытой координацией $K_{скрыт}$.

5. Признак "характер контактов". Типы проектов:

- ✓ внутренние контакты $X_{внутр}$;
- ✓ региональные контакты $X_{регион}$;
- ✓ международные контакты $X_{м\wedge нар}$.

6. Признак "количество участников проекта". Типы проектов:

- ✓ индивидуальные $Y_{инд}$;
- ✓ парные $Y_{парн}$;
- ✓ групповые $Y_{групп}$.

7. Признак "продолжительность проекта". Типы проектов:

- ✓ краткосрочные $P_{крат}$;
- ✓ средней продолжительности $P_{сред}$;
- ✓ долгосрочные $P_{долг}$.

Определяя общий классификационный признак проекта ОКПП (в обозначениях БНФ – форм Бэкуса-Наура) получим:

ОКПП ::= M C F R K X Y P.

M ::= M_{иссл} | M_{твор} | M_{инф} | M_{тв-инф} | M_{инф-иссл} | M_{тв-иссл} | M_{тв-инф-иссл}.

C ::= C_{уч} | C_{уч-прикл}.

F ::= F_{рут} | F_{кв-рут} | F_{нерут}.

K ::= K_{откр} | K_{скрыт}.

X ::= X_{внутр} | X_{регион} | X_{м-нар}.

Y ::= Y_{инд} | Y_{парн} | Y_{групп}.

P ::= P_{крат} | P_{сред} | P_{долг}.

Приведем примеры некоторых из них с указанием автора.

Пример 1: "Internet-газета" ($M_{тв-инф} C_{уч-прикл} F_{кв-рут} K_{откр} X_{внутр} Y_{групп} P_{крат}$). Используя язык разметки гипертекста HTML, язык JavaScript и в случае необходимости – Java, создать Internet-газету факультета Информатики, имеющей следующие обязательные разделы: "Новости", "Это интересно", "Поздравляем", "Объявления" и по своему усмотрению. Мате-

риалы для объявлений и новостей взять в деканате факультета. Авторский коллектив: Неустроева Ю., Титова И., Хасиев Р., Кириллов А.

Пример 2: «Игра «Балда»». ($M_{\text{те-иссл}}$ $C_{\text{уч}}$ $F_{\text{кв-рут}}$ $K_{\text{откр}}$ $X_{\text{внутр}}$ $Y_{\text{инд}}$ $P_{\text{сред}}$). Суть данного проекта заключается в создании интерактивной среды, в которой студент имеет возможность самостоятельно развивать интеллектуальные навыки, повышать свой словарный запас и т.д. Язык программирования – Turbo Pascal. Авторский коллектив: Тигиняну Е., Федотова Е., Попов Е., Лукин А., Уваров С.

1.3. Организация технологического процесса обучения программированию на основе метода проектов

Методика обучения с применением любой педагогической технологии строится на основе определения ее технологической карты и технологической схемы.

Технологическая схема – условное изображение технологии процесса, разделение его на отдельные функциональные элементы и обозначение логических связей между ними (Рис. 3).

Технологическая карта – описание процесса в виде пошаговой, поэтапной последовательности действий (часто в графической форме с указанием применяемых средств (Рис. 4).

Процесс обучения, исходя из технологической схемы метода проектов, строится следующим образом:

1. Исходным техническим заданием для учителя являются требования к усвоению обучаемым определенного набора знаний, умений и навыков, предусмотренных учебной программой, развития познавательных потребностей учащегося и т.д. Техническим заданием для ученика является разработка учебного программного проекта.

2. Учитель совместно с обучаемыми создает проблемную ситуацию, для разрешения которой требуется разработать и реализовать программный проект определенной тематики.

3. Ученик получает (на первых этапах), разрабатывает самостоятельно (либо с помощью учителя) спецификацию программного проекта. Проект разбивается на подзадачи. Результат этой работы проверяется учителем или экспертной группой.

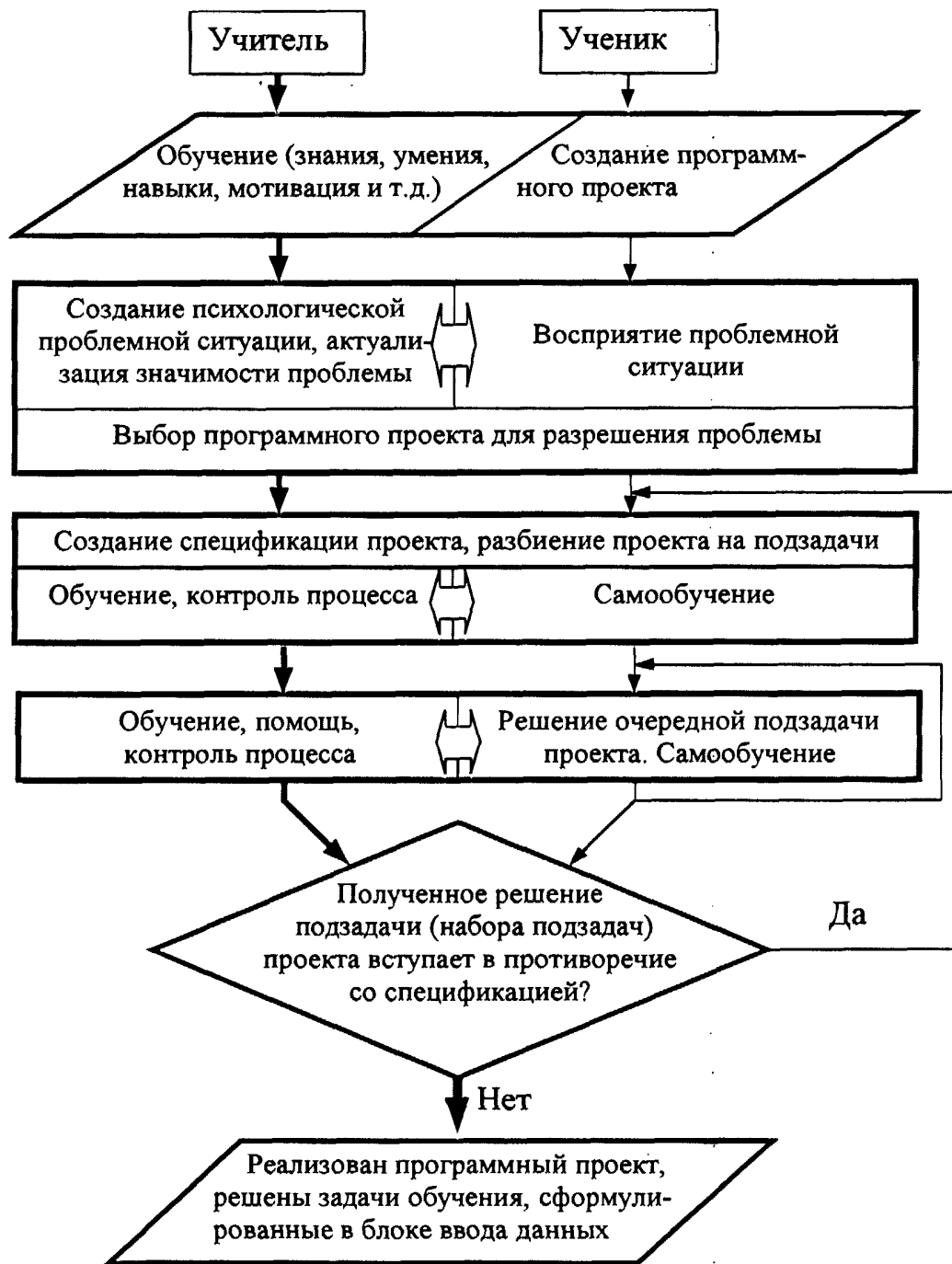


Рис. 3. Технологическая схема обучения по методу проектов

4. Предпосылками к решению отдельной подзадачи, а также ее следствием является усвоение и применение знаний, умений и навыков, получаемых в результате обучения и самообучения школьников. Результат ре-

шения подзадачи проверяется учителем или экспертной группой. В случае несоответствия полученного решения спецификации проекта, происходит возврат на пункт 3.

5. Для решения следующей подзадачи производится возврат на пункт 4.

6. Результатом совместной деятельности учителя и ученика является достижение всех целей, определенных в пункте 1.

Мы выделили ряд этапов процесса обучения с использованием метода проектов, которые необходимо пройти каждому обучаемому.

Каждый этап в предлагаемой модели процесса обучения обладает определенными целями, содержанием и процессуальными характеристиками, направленными в конечном итоге на формирование высокого уровня владения знаниями, умениями и навыками программирования. Некоторые функции этапов, их дидактические характеристики, а также содержание эксплицированы в предлагаемой обобщенной динамической модели.

1 этап. Исходно-установочный.

Цель: Организация проблемной ситуации, ее интериоризация. Выбор проекта, подготовка его спецификации.

Содержание этапа:

- ❖ организация проблемной ситуации, ее интериоризация. Выбор проекта, реализация которого разрешает рассматриваемую проблему;
- ❖ демонстрация образца проекта, желательного обладающего минимумом функциональных возможностей (этот пункт содержания этапа необязателен);
- ❖ создание либо представление спецификации проекта в общем виде

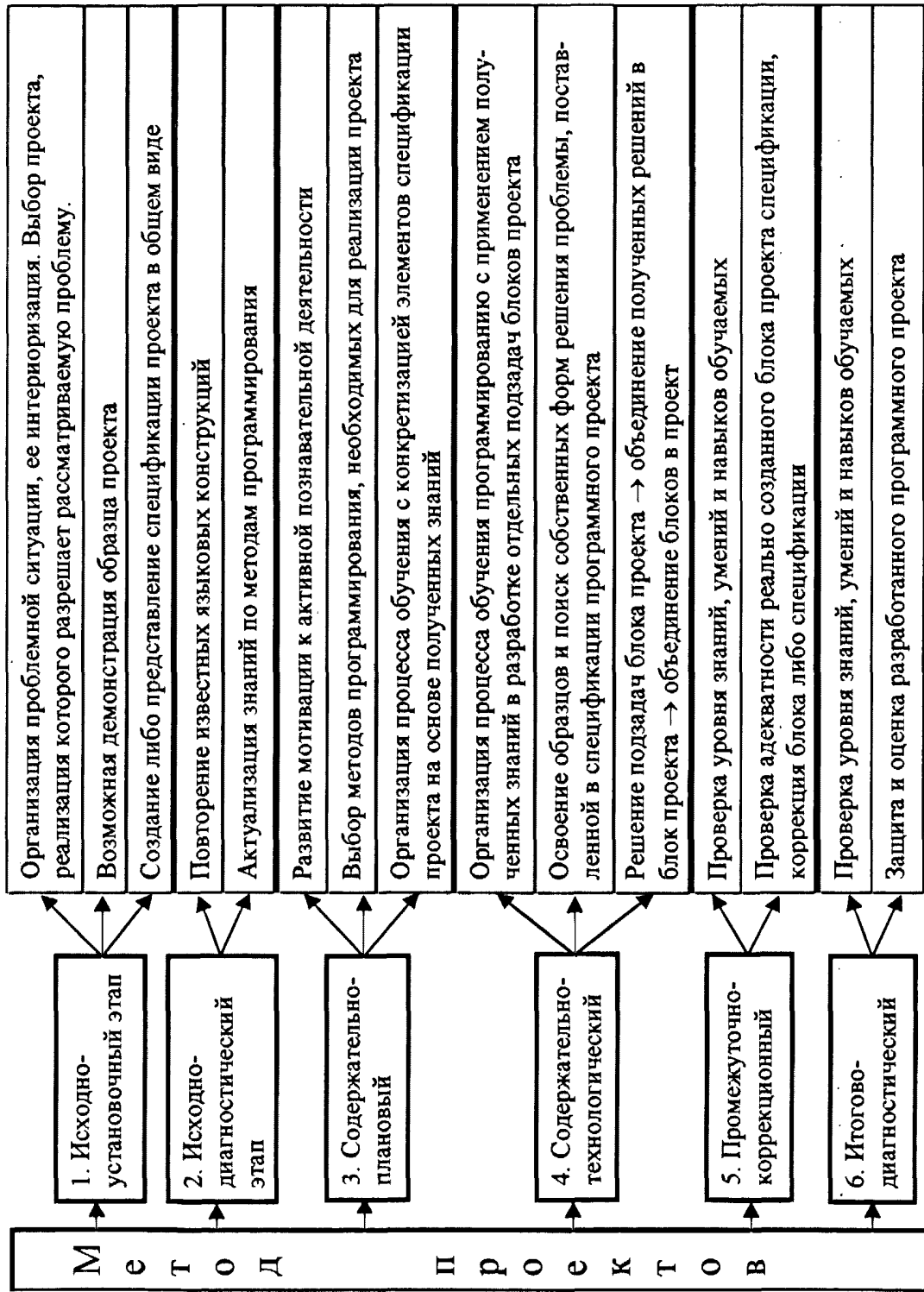


Рис. 4. Технологическая карта процесса обучения программированию методом проектов

2 этап. Исходно-диагностический

Цель: *актуализация имеющихся знаний по языкам и методам программирования, необходимых для разработки проекта.*

Содержание этапа:

- ❖ повторение известных языковых конструкций;
- ❖ актуализация знаний по методам программирования.

3 этап. Содержательно-плановый

Цель: рациональное и логическое осмысление основ изучаемого языка (лексемы, структура программы, типы данных), изменение соответствующего функционально возможного для обучаемого пространства и мотивации учебно-познавательной деятельности.

Содержание этапа:

- ❖ развитие мотивации к активной познавательной деятельности с целью совершенствования процесса разработки проекта.
- ❖ определение знаний, умений и навыков, требуемых для создания блока проекта, и выбор оптимальных методов программирования, необходимых для его реализации;
- ❖ организация процесса обучения с конкретизацией элементов спецификации проекта на основе полученных знаний;

Этот этап предполагает следующие формы и методы организации деятельности обучаемых: мотивационное обеспечение процесса, научное и логическое обоснование основных идей программирования, выделение структурных элементов, составление схем отдельных компонент теоретического материала, сравнение полученных схем с уже имеющимися, диалоговая форма построения занятий (групповые дискуссии, групповые формы проектов и т.д.), тренинг программирования на базе имеющихся знаний.

4 этап. Содержательно-технологический.

Цель: Реализация подзадач проекта, объединение их в блоки. Использование новых знаний в создании больших программных проектов, выделение новых методов программирования.

Содержание этапа:

- ❖ организация процесса обучения программированию с применением получаемых знаний в разработке отдельных подзадач блоков проекта, формирование нового понятийного аппарата;
- ❖ освоение образцов и поиск собственных форм решения проблемы, поставленной в спецификации программного проекта;
- ❖ решение подзадач блока проекта, объединение полученных решений в блок проекта, объединение блоков в проект.

Данный этап предполагает использование следующих форм и методов организации деятельности обучаемых: организация сотрудничества между обучаемыми в рамках одного проекта, предъявление новых структурных компонентов языка, методов и форм программирования, создание ситуаций выбора между старыми и новыми приемами программирования, показ и совместный поиск вариантов построения функций (процедур, подпрограмм, классов и т.д.), направленных на реализацию или поддержку реализации некоторого действия и т.д..

5 этап. Промежуточно-коррекционный.

Цель: Промежуточный контроль и корректировка процесса с возможностью возврата на любой из предыдущих этапов. Контроль усвоения знаний, умений и навыков.

Содержание этапа:

- ❖ проверка уровня знаний, умений и навыков обучаемых;
- ❖ проверка адекватности реально созданного блока проекта спецификации, коррекция блока либо спецификации;

6 этап. Итогово-оценочный.

Цель: Итогово-оценочный контроль реализованного программного проекта. Контроль усвоения знаний, умений и навыков.

Содержание этапа:

- ❖ проверка уровня знаний, умений и навыков обучаемых;
- ❖ защита и оценка разработанного программного проекта.

Технологические схема и карта позволяют рассматривать обучение основам программирования с использованием метода проектов как некоторую педагогическую процессуальную систему, обладающую воспроизводимостью при условии следования технологическим требованиям, контролируруемую на каждом технологическом этапе.

Выводы по главе 1:

1. Для создания методики обучения программированию в курсе информатики, основанной на применении метода проектов необходимо разработать соответствующую педагогическую технологию, обеспечивающую возрастание познавательных потребностей обучаемых и повышение эффективности процесса обучения.

2. Учебным проектом будем называть проблемную задачу, процесс анализа и решения которой может быть алгоритмизован, обладает свойствами поэтапности и длительности, опирается на систему научных знаний в изучаемой области.

3. Методом проектов будем называть педагогическую технологию, основанная на широком внедрении в процесс обучения учебных проектов, обладающих социальной и личностной значимостью.

4. К типологическим признакам учебных проектов следует отнести: цели создания проекта, методы, доминирующие в процессе разработки проекта, характер творческой деятельности при разработке проекта, харак-

тер координации проекта, характер контактов и количество участников проекта, продолжительность разработки проекта.

5. Поскольку структура педагогической технологии базируется на использовании технологической карты и технологической схемы процесса обучения, то для разработки технологии обучения, соответствующей методике применения метода проектов, необходимо наполнение блоков технологической схемы и этапов технологической карты содержанием, отражающим специфику метода проектов в обучении программированию в курсе информатики.

6. Анализ разработанной технологической схемы процесса обучения, основанного на использовании метода проектов, показали алгоритмизованность, поэтапность и цикличность данного процесса.

7. К этапам процесса обучения программированию на основе использования метода проектов, выделенным в технологической карте, относятся: исходно-установочный; исходно-диагностический; содержательно-плановый; содержательно-технологический; промежуточно-коррекционный; итогово-оценочный, каждому из которых соответствуют свои цели и содержание.

ГЛАВА 2. МЕТОДИКА ИСПОЛЬЗОВАНИЯ МЕТОДА ПРОЕКТОВ ПРИ ОБУЧЕНИИ ПРОГРАММИРОВАНИЮ В СТАРШИХ КЛАССАХ ОБЩЕОБРАЗОВАТЕЛЬНОЙ ШКОЛЫ

Вторая глава посвящена методике применения метода проектов в средней школе, а также принципам подготовки учителей информатики к использованию методов проектов. Рассматривается роль методов проектов в развитии познавательных потребностей школьников, а также особые возможности больших программных проектов в решении данной психолого-педагогической проблемы. Раскрыто понятие спецификации программного проекта, особое внимание уделено принципам построения спецификаций проектов в средней школе. Выделяются уровни готовности обучаемых к проектной деятельности и определяется методика реализации технологических этапов метода проектов с учетом выделенных уровней. Проанализированы принципы формирования готовности будущих учителей информатики к использованию метода проектов в своей профессиональной деятельности.

2.1. Роль метода проектов в развитии познавательных потребностей школьников в процессе обучения программированию

Одним из важнейших условий успешности процесса обучения школьников является создание педагогических ситуаций, в результате реализации которых учащиеся осознают необходимость и важность получаемых знаний. Это прямым образом влияет на создание положительной мотивации учения, что, в свою очередь, ведет к возникновению познавательных потребностей (ПП). Остановимся подробнее на процессе формирования ПП при использовании в обучении метода проектов.

Процесс формирования потребностей приобретения знаний следует рассматривать с двух позиций: с одной стороны, потребности – это психическое состояние личности; с другой, – это компонент мотивационной сферы обучаемого.

Потребность как состояние личности всегда связана с наличием у человека чувства неудовлетворенности, связанного с дефицитом того, что требуется организму. Личность избирательно реагирует на то, что составляет предмет потребности. Потребность активизирует организм, стимулирует его поведение, направленное на поиск того, что требуется.

Основными характеристиками человеческих потребностей считаются: сила, периодичность возникновения и способ удовлетворения. Дополнительной характеристикой является предметное содержание потребности, т.е. совокупность тех объектов материальной и духовной культуры, с помощью которых данная потребность будет удовлетворена. В случае познавательной потребности ее предметным содержанием являются знания, а способом удовлетворения – процесс познания [118, с.465].

Формирование и развитие потребностей подчиняется нескольким правилам.

1. Потребность может возникнуть и укрепиться только в процессе систематического осуществления определенной деятельности, что обеспечивает привыкание организма и личности в целом к определенному роду функционирования. Все новые потребности формируются по типу привычки; одинаков и их механизм функционирования.

2. Потребность развивается при условии ее расширенного воспроизводства, что обеспечивается разнообразием и развитием средств ее удовлетворения. Потребность возникает и развивается в процессе деятельности.

3. Становление потребности облегчается, когда деятельность, особенно вначале, относительно легко осуществляется, что обеспечивает положительное отношение к ней личности.

4. Важным условием развития потребностей является переход от репродуктивной к творческой деятельности, что не только укрепляет положительное эмоциональное отношение к ней личности, но и ведет к признанию данной деятельности основным признанием личности.

5. Потребность укрепляется при сознании ее личной и общественной значимости.

Эти правила находят подтверждение в создании ПП на основе использования метода проектов. Первое правило обосновывается необходимостью постоянного поиска путей решения поставленной задачи, поиска или актуализации необходимого теоретического материала. Деятельность, направленная на реализацию проекта, и является средством удовлетворения ПП (второе правило). Первоначальные шаги в создании проекта, выполненные под руководством учителя или самостоятельно, легко выполнимы, что стимулирует обучаемого, "втягивает" его в процесс реализации проекта (третье правило). Репродуктивные методы в большом программном проекте используются только на первых этапах, когда разработанная учителем первоначальная модель проекта не претерпевает значительных изменений, а также при реализации хорошо известных алгоритмов. На последующих этапах, при росте количества отклонений от исходной модели, особенно в ситуациях, когда учитель не ограничивает обучаемого жесткими рамками формы представления результатов, четко и однозначно определяя только минимальный уровень требований, на первый план выходит творческий подход как к решению тех или иных задач проекта, так и к проекту в целом. Как следствие, у разных групп обучаемых проекты отличаются интерфейсом и удобством в управлении, разнообразием дополнительных возможностей, скоростью выполнения и объемом кода и т.д. (четвертое правило). И, наконец, последнее (пятое) правило находит свое отражение, во первых, в самом методе проектов как деятельности, наиболее близкой по форме и структуре к профессиональной деятельности програм-

миста, а во вторых, в тематике создаваемого проекта, которая при должном подходе учителя к данному вопросу основана на текущих нуждах обучаемого, класса, школы, студенческой группы, института и т.д.

Анализ процесса формирования потребностей в приобретении знаний или познавательных потребностей (ПП) следует начать с рассмотрения основных характеристик потребностей: сила, периодичность возникновения, способ удовлетворения и предметное содержание (Таблица 4).

Рассмотрим далее потребность как компонент мотивационной сферы. Создание потребностей в приобретении знаний – сложный процесс, основанный на изучении мотивации познавательной деятельности и структурного анализа ее целей.

Мотивационная сфера человека состоит из следующих мотивационных образований: диспозиции (мотивы), потребности, цели.

С точки зрения развитости мотивационную сферу можно оценить по следующим параметрам: широта, гибкость и иерархизированность.

Под широтой мотивационной сферы понимают качественное разнообразие мотивационных факторов – диспозиций (мотивов), потребностей и целей, представленных на каждом уровне [118, с. 466]. Широта мотивационной сферы в области мотивов учения и, конкретнее, потребности приобретения знаний, будет рассматриваться как разнообразие целей.

Гибкость мотивационной сферы характеризует процесс мотивации следующим образом. Более гибкой считается такая мотивационная сфера, в которой для удовлетворения мотивационного побуждения более общего характера (более высокого уровня) может быть использовано больше разнообразных мотивационных побудителей более низкого уровня [118, с. 467].

Таблица 4

Сравнительная характеристика реализации условий формирования познавательных потребностей при традиционном обучении и при использовании метода проектов

Условия	Реализация условий при традиционном обучении	Реализация условий при использовании метода проектов
<p>1. Процесс обучения построен на основе нехватки знаний для решения представленной проблемы</p>	<p>Проблемный подход используется при введении новых понятий, свойств, требует значительной подготовки при разработке проблемы, решение которой поможет в изучении той или иной темы (констатирующий эксперимент показал, что 85% учителей информатики, работающих по традиционной системе, не используют проблемный подход в обучении)</p>	<p>СИЛА ПОТРЕБНОСТИ</p> <p>Метод проектов ЗАСТАВЛЯЕТ использовать проблемный подход в обучении, которой является одним из важнейших его частей. При использовании этого метода перед учащимися ставится проблема, разрешается на несколько подпроблем, в процессе разрешения которых обучаемый получает новые знания об изучаемом объекте. (Эксперимент показал, что 100% учителей, работающих по методу проектов осознанно и целенаправленно применяют проблемный подход в обучении, причем 72% обучаемых указали на то, что они с большим интересом сами узнают что-то новое, "догадываются", чем их учитель "все подносит на блюдечке")</p>

Условия	Реализация условий при традиционном обучении	Реализация условий при использовании метода проектов
2. Осознание обучаемыми необходимости получения знаний	<p style="text-align: center;">СИЛА ПОТРЕБНОСТИ</p> <p>В традиционных методах обучения, где основным методом закрепления знаний и получения основных умений и навыков является решение небольших задач и выполнение упражнений, достаточно сложно показать практическую полезность приобретаемых знаний, трудно выделить их прикладную направленность</p>	<p>При использовании метода проектов в обучении основное внимание уделяется актуальности и практической направленности решаемой проблемы. Таким образом, метод проектов позволяет обучаемым познать прикладной характер получаемых знаний, полезность приобретаемых умений и навыков</p>
3. Большая доля самостоятельной работы в процессе обучения	<p>В традиционном обучении учителю отводится роль основного, ведущего источника или посредника в получении знаний. Результатом такого подхода часто является низкая самостоятельность обучаемых в процессе познания, неподготовленность к реальным жизненным ситуациям, возникающим в будущей профессиональной деятельности</p>	<p>При использовании метода проектов учитель приобретает статус первого, но не единственного помощника в реализации проекта. Он перестает быть беспрекословным авторитетом в изучаемой области, получаемые от него знания стоят на одном уровне с полученными самостоятельноным исследованием. В предельном случае ученик может самостоятельно ставить перед собой задачи очередного этапа познания, искать пути их решения, пытаться представить необходимый результат и формулировать вопросы учителю, ответы на которые ему необходимы для решения проблемы</p>

Условия	Реализация условий при традиционном обучении	Реализация условий при использовании метода проектов
Спиралевидный характер процесса обучения	Традиционная система знаний по программированию строится на основе изучения всей теории в одной плоскости	Структура процесса обучения по методу проектов циклична и имеет спиралевидный характер (Рис. 8)
СПОСОБ УДОВЛЕТВOREНИЯ ПОТРЕБНОСТИ (ПОЗНАНИЕ)		
Осознанность процесса познания	Традиционное обучение не позволяет обучаемым в достаточной мере осознать важность методологического аспекта процесса познания	Обучаемый полностью осознает необходимость вызовов осуществления процесса познания
ПРЕДМЕТНОЕ СОДЕРЖАНИЕ ПОТРЕБНОСТИ (знания, умения и навыки)		
Выявление четкой взаимосвязи между отдельными компонентами знаний, умений и навыков	Раскрытие взаимосвязи между отдельными компонентами знаний, умений и навыков осуществляется в подавляющем большинстве случаев с помощью учителя, доля самостоятельной работы учащегося в этом процессе невелика.	Метод проектов позволяет естественным образом акцентировать внимание обучаемого на связях между отдельными компонентами знаний, на возможности трансформации умений и навыков из одного вида в другой

Заметим, что широта и гибкость характеризуют мотивационную сферу человека по-разному. Широта – это разнообразие потенциального круга предметов, способных для данного человека служить средством удовлетворения актуальной потребности, а гибкость – подвижность связей, существующих между разными уровнями иерархичной организованности мотивационной сферы: между мотивами и потребностями, мотивами и целями, потребностями и целями.

Иерархичность – это характеристика строения каждого из уровней организации мотивационной сферы, взятого в отдельности. Потребности, мотивы, цели не существуют как рядопологаемые наборы мотивационных диспозиций. Одни диспозиции (мотивы, цели) сильнее других и возникают чаще; другие слабее и актуализируются реже. Чем больше различий в силе и частоте актуализации мотивационных образований определенного уровня, тем выше иерархизированность мотивационной сферы.

Поведение, направленное на удовлетворение потребностей, разделяется на виды деятельности (общения), соответственные частным целям.

Анализ потребностей как компонента мотивационной сферы показывает большую роль целей в формировании познавательных потребностей.

Целеобразование – процесс порождения новых целей в деятельности человека, одно из проявлений мышления [135, с. 440].

Компоненты целеобразования реализуются следующими механизмами:

- превращение полученных требований в индивидуальную цель;
- выбор одного из имеющихся требований;
- превращение мотивов в мотивы-цели при их осознании;
- превращение побочных результатов действия в цель;
- преобразование неосознанных предвосхищений в цель;
- выделение промежуточных целей;
- переход от предварительных к окончательным целям;

- образование иерархии и временной последовательности целей [135, с. 440].

При традиционном обучении процессом целеобразования занимается учитель. Он должен так построить обучение, чтобы из системы целей на данный урок можно поставить цель развития познавательных потребностей. При этом деятельность учителя усложняется. Использование метода проектов в обучении предполагает автоматическое использование системы целей, направленных на развитие ПП (Рис. 5).

Под компонентом проекта будем понимать ограниченный временными или смысловыми рамками этап реализации проекта.

Под целеобразовательным компонентом будем понимать этап реализации проекта, способствующий самостоятельному образованию у обучаемых целей формирования познавательных потребностей.

Определим каждый из представленных этапов и выделим механизмы целеобразования на каждом из них.

1. Целеобразовательный компонент, направленный на выделение задач необходимости владения определенными личностными качествами, предполагает образование целей по установлению социальных потребностей. Они выражаются в установлении сотрудничества в группе и определяют необходимость приобретения знаний для адаптации в коллективе.

Механизмы целеобразования:

- распределение обязанностей среди членов группы;
- осуществление контроля над каждым человеком;
- коммуникативные связи между членами группы.

При этом формируются коммуникативные умения, умения совместной деятельности, распределять роли в группе и т.д.

2. Целеобразовательный компонент проекта, направленный на выделение задач формирования прикладных умений опирается на осознание необходимости практических навыков программирования.

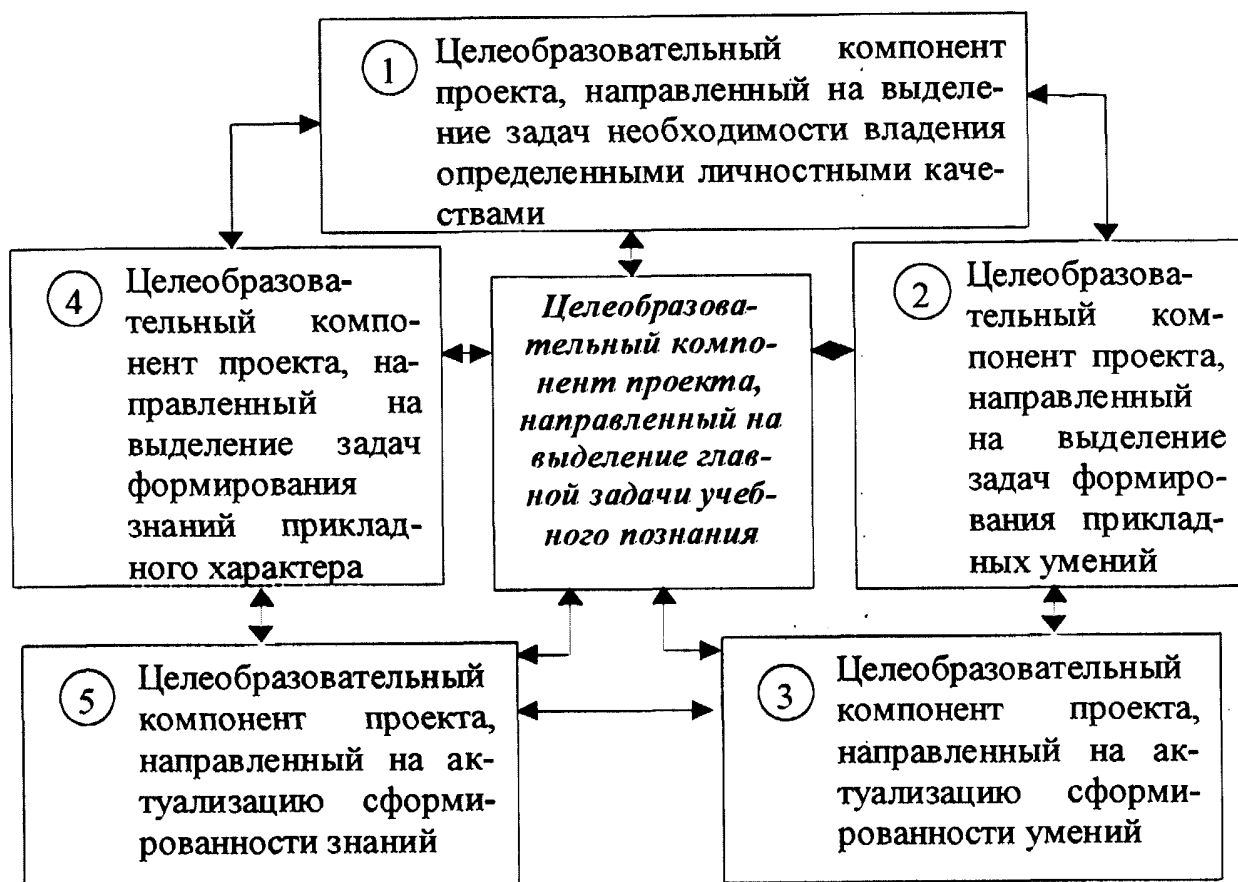


Рис. 5. Комплекс компонентов проекта, используемых в процессе целеобразования.

Данный компонент реализует образование целей, направленных на развитие познавательных потребностей, связанных с прикладными возможностями программирования. При реализации проекта происходит выделение промежуточных целей, среди которых особую роль играют следующие:

- формирование умений оптимального применения языковых конструкций;
- развитие умений программирования простейших нечисловых алгоритмов, например, поиска и сортировки;
- умения применять изученные алгоритмы для работы с нестандартными типами данных;

- умения программировать простейший интерфейс с различными видами устройств ввода данных (клавиатура, мышь, джойстик и т.д.).

Вышеперечисленные умения опираются, в свою очередь, на умения более низкого уровня, как то:

- умения программировать линейные, разветвленные и циклические алгоритмы;

- умения использовать графические возможности языка программирования;

- умения использовать возможности среды программирования при создании алгоритмов и т.д.

3. Целеобразовательный компонент проекта, направленный на актуализацию сформированности умений. Каждое из сформированных умений неоднократно актуализируется в процессе создания программного проекта. Структуру взаимоотношений простейших (ПУ), сложных (СУ) и более сложных (БСУ) умений можно представить в виде гомоморфных отношений множеств:

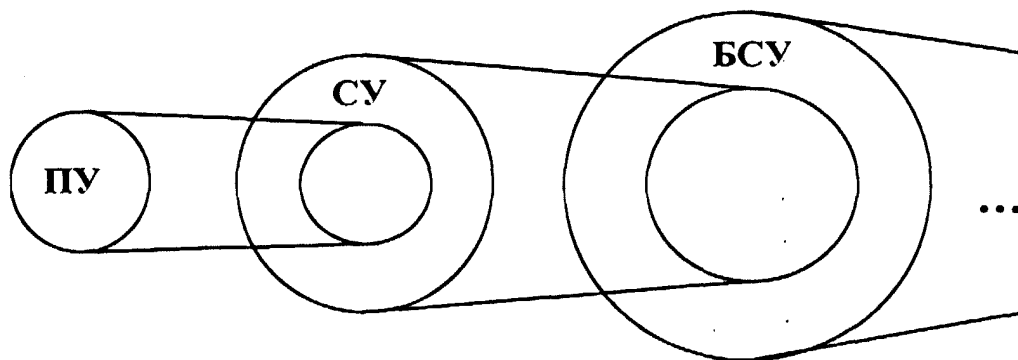


Рис. 6. Структура гомоморфных отношений формируемых в процессе разработки проекта умений

Таким образом, мы приходим к образованию иерархии или временной последовательности целей: формирование ПУ → формирование на их основе СУ → формирование БСУ → ...

4. Целеобразовательный компонент проекта, направленный на выделение задач формирования знаний прикладного характера опирается на представления обучаемых о возможности и необходимости использования полученных знаний при дальнейшей разработке программного проекта. Данный компонент играет одну из важнейших ролей в формировании познавательных потребностей. Механизм целеобразования в данной ситуации строится на основе превращения мотивов в мотивы – цели при их осознании. Основным мотив в данном случае – это прикладной характер осуществляемой деятельности, что дает осознание практической значимости результатов реализации проекта.

5. Целеобразовательный компонент проекта, направленный на актуализацию сформированности знаний основывается на необходимости в дальнейшей учебной (профессиональной) деятельности получаемых знаний. Механизм целеобразования строится аналогично п. 3.

Выделенные целеобразовательные компоненты иллюстрируют свойство широты мотивационной сферы.

Гибкость же в случае обучения с помощью метода проектов характеризуется разнообразием связи между мотивами и потребностями, мотивами и целями, потребностями и целями. Это выражается в самой структуре процесса, в котором хорошо представлена гибкость мотивационной сферы в момент "качественного" скачка, перехода от одной стадии проекта к другой.

Таким образом, использование МП в обучении предполагает наличие следующих условий развития познавательных потребностей:

- 1) наличие положительной мотивации;
- 2) четкая формулировка целей (целеполагание);
- 3) непрерывность процесса обучения;
- 4) большая доля самостоятельной работы теоретического и практического характера;

5) спиралевидно-циклический характер приобретения знаний.

Процесс реализации учебного проекта включает в себя данные условия в качестве отдельных компонентов (Рис. 7, Рис. 8).

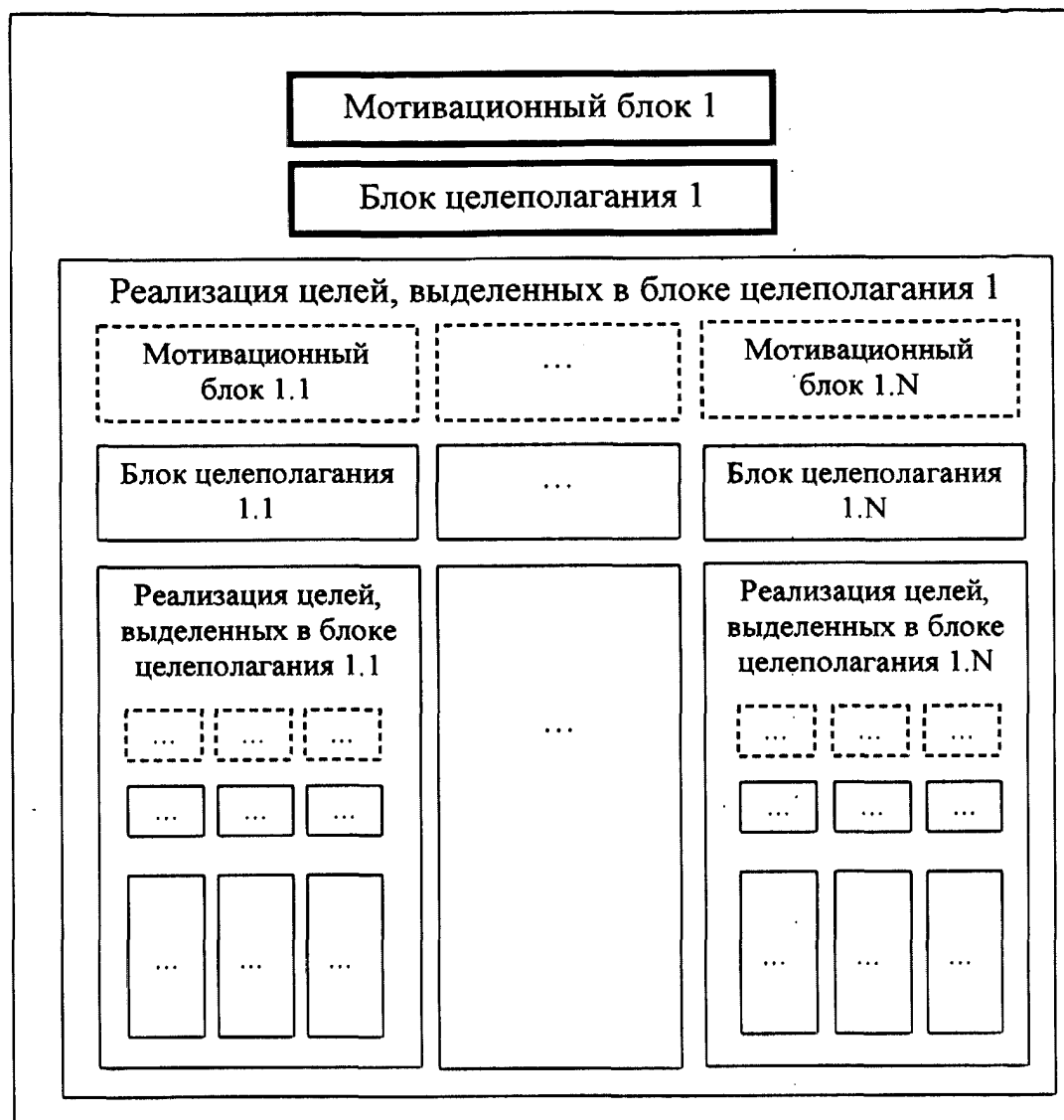


Рис. 7. Диаграмма процесса реализации учебного проекта с точки зрения развития мотивационной сферы.

Диаграмма (Рис. 7) в полной мере показывает иерархизованность и цикличность процесса реализации учебного проекта. Наиболее важным с точки зрения развития мотивационной сферы является первичный мотивационный блок, инициируемый учителем. Здесь определяется актуальность

данного проекта, его важность, востребованность, личная и социальная значимость, формируется потребности и мотивы деятельности по реализации проекта. Первичный блок целеполагания на начальных этапах обучения по методу проектов определяется учителем, роль которого впоследствии в данном блоке переходит из директивной формы в аналитическую. Здесь производится определение наиболее общих целей, следствием достижения которых является реализация проекта.

Блок реализации целей подразумевает последовательное (один обучаемый) или последовательно-параллельное (несколько обучаемых) достижение выделенных целей, причем данный блок может иметь структуру, аналогичную структуре основного. Особенностью внутренних блоков реализации целей является упрощение методов формирования соответствующих мотивационных блоков. Это компенсирует трудоемкость организации внешнего мотивационного блока, и является следствием сформированности основных целей, выделенных в первичном блоке целеполагания.

Рассмотрим более подробно процесс реализации целей учебного программного проекта (Рис. 8).

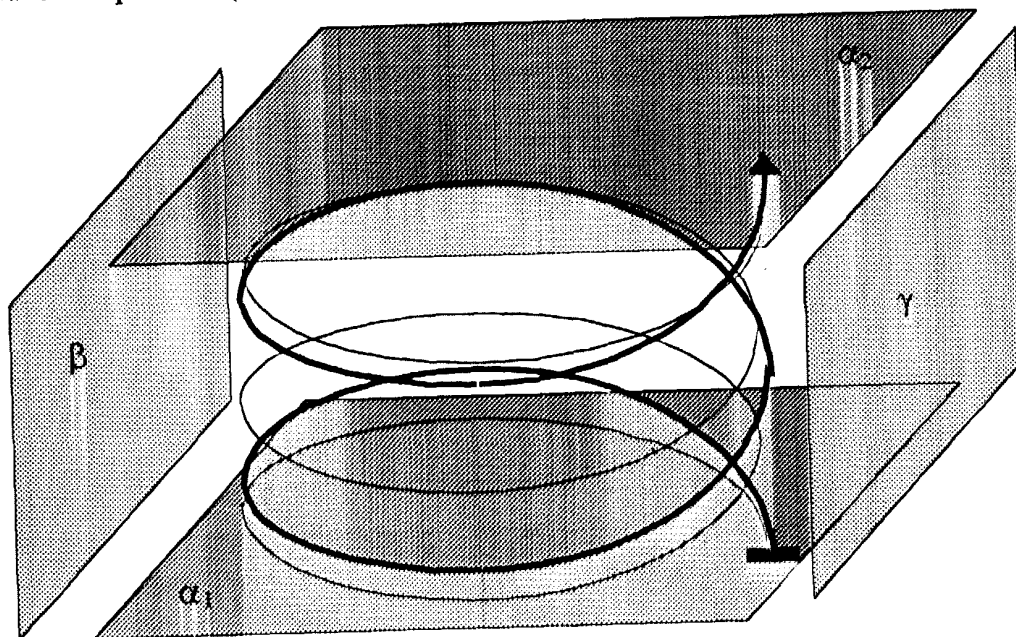


Рис. 8. Модель процесса реализации целей программного проекта

На схеме представлена модель, состоящая из двух параллельных плоскостей α_1 и α_2 , означающих уровни качественного перехода от одной стадии разработки проекта к другой. Этот переход подразумевает приобретение обучаемыми соответствующих знаний, умений и навыков, с помощью которых и реализован данный структурный компонент проекта.

Плоскости β и γ соответствуют теоретическому и практическому этапам реализации проекта.

Пространство, заключенное между плоскостями α_1 и α_2 назовем информационным блоком, тогда расстояние между этими же плоскостями обозначает уровень информационной компетентности этапа. Он включает в себя теоретический материал, перечень умений и навыков, которые должны быть усвоены учащимся по завершении блока. Будем обозначать эту величину T и измерять в единицах информации.

$$T=d(\alpha_1, \alpha_2)$$

Представленная на модели спираль используется для наглядного представления траектории реализации проекта.

Каждый виток спирали предполагает наличие следующих блоков реализации процесса приобретения знаний, умений и навыков при создании проекта:

БЛОК 1. Создание виртуальной модели.

- 1.1 Определение объектов, свойств и их характеристик, необходимых для успешного освоения данного этапа.
- 1.2 Определение формы представления (типы данных: массивы, записи, объекты и т.д.).
- 1.3 Определение методов представления (какие требуются процедуры и функции, методы объектов и т.п.).

БЛОК 2. Реализация модели.

2.1 Реализация модели с проверкой на каждом этапе (предварительная компиляция, создание отладочных функций, реализация промежуточных объектов и т.д.).

В случае неадекватности полученной реальной модели виртуальной обучаемый начинает новый виток спирали в том же информационном блоке с уточнения, расширения, модификации исходной модели с ее последующей реализацией.

Таким образом, освоение обучаемым единичного информационного блока можно представить в виде блок-схемы (Рис. 9).



Рис. 9. Блок-схема освоения обучаемым информационного блока

Плоскостей α может быть несколько, т.е. $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_m\}$, соответственно и подпространств (информационных блоков) может быть $m-1$ ($\Pi_1, \Pi_2, \dots, \Pi_{m-1}$). Обозначим через T_i расстояние между плоскостями α_i и α_{i+1} .

T_i - это функция, зависящая от сложности этапа, корректности формулировки задания, объема работы, которую необходимо выполнить для реализации проекта, т.е.

$$T_i = t_i (\text{сложность, корректность, объем}) \quad (1)$$

В подпространство Π_i , заключенное между плоскостями α_i и α_{i+1} , может быть заложен один виток спирали (это означает, что программный модуль, реализующий данный компонент проекта, полностью удовлетворяет всем требованиям); или несколько (следовательно, при создании допущены ошибки, которые приводят к пересмотру модели в целом).

Введем величину, зависящую от количества витков в подпространстве Π_i . Назовем ее *информационной компетентностью витка* и будем вычислять ее по формуле:

$$\rho_i = \frac{T_i}{n_i},$$

где n_i - количество витков спирали в подпространстве Π_i .

Информационная компетентность витка указывает на уровень развития мышления обучаемых. Чем выше информационная компетентность витка, тем выше уровень усвоения и владения необходимыми знаниями, умениями и навыками (прямо пропорциональная зависимость). По данной формуле виток будет обладать максимальной информационной компетентностью при $n_i=1$. Тем не менее, как показало наше исследование, не следует стремиться к максимальной информационной компетентности каждого витка спирали, так как эффективность дальнейшего обучения (в подпространствах $\Pi_{i+1} \dots \Pi_{m-1}$) значительно повышается при $n_i=2$. Это озна-

чает, что процесс осуществления некоторой стадии проекта состоит из двух этапов – первоначального, когда создается и реализуется исходная модель, и заключительного, когда осознаются недостатки исходной модели, и происходит ее теоретическая и практическая переработка в соответствии с результатами первоначального этапа. Такой подход, с одной стороны, в достаточной степени моделирует реальный процесс научного познания, а с другой, позволяет не потерять интерес к программному проекту в результате слишком частого изменения и уточнения исходной модели.

Таким образом, ρ_i для каждого обучаемого вычисляется индивидуально и исходя из этого делаются выводы о сформированности знаний, умений и навыков, о развитии мышления.

Из определения ρ_i следует, что T_i имеет более сложную зависимость:

$$T_i = t_i(\text{сложность, корректность, объем, } \rho_{i-1}) \quad (2)$$

Таким образом, использование метода проектов позволяет приблизить процесс обучения к научно-исследовательской деятельности, что положительно сказывается на создании познавательных потребностей. Исходя из разбиения всей структуры проекта на блоки, автоматизируется процесс представления процесса обучения в виде подпространств. Сложность в данном случае представляет количественное определение информационной компетентности каждого витка, с учетом ее зависимости от информационной компетентности предыдущих витков. На сегодняшний день данная проблема решена только в качественной области и нуждается в дополнительных исследованиях.

Реализация программного проекта, который строится по данной конструкционной схеме, наглядно представляет свойство непрерывности процесса и его спиралевидный характер. Та же самая схема, как показывает опыт, активно используется обучаемыми по методу проектов для организации своего собственного процесса познания.

Исходя из вышеизложенного, можно сделать следующие выводы:

1. Потребность в приобретении знаний в области программирования имеет практически постоянный характер на витке спирали, но резко усиливается в период перехода к новому витку или функциональному состоянию (стадии).

2. Исходя из структурной сложности любого профессионального языка программирования, можно сделать вывод о повышении эффективности обучения программированию при условии, что данный процесс построен по принципу постоянного усложнения и увеличения объема знаний на каждой последующей стадии:

$$T_1 \leq T_2 \leq T_3 \leq T_4 \leq \dots$$

причем

$$T_2 - T_1 \geq T_3 - T_2 \geq T_4 - T_3 \geq \dots$$

3. Построение процесса обучения по методу проектов в соответствии с предложенной моделью позволяет в полной мере реализовать условия развития познавательных потребностей школьника.

2.2. Принципы построения спецификаций программных проектов в общеобразовательной школе

Необходимым условием успешности процесса целеобразования при использовании метода проектов в обучении является четкое, детальное и иерархизованное представление целей учебного проекта, выраженное в письменной форме. Другим, не менее важным условием является наличие поэтапного планирования действий, служащих достижению выделенных целей. Оба условия реализуются в *спецификации* программного проекта.

Под спецификацией проекта будем понимать его подробное описание с выделением основных компонентов процесса планирования и реализации. Спецификация включает в себя как предполагаемый результат деятельности (цель), так и подробный план (с зависимостью детализации от уровня готовности учащихся к проектной деятельности) достижения искомого результата. Для метода проектов в обучении программированию спецификация предполагает не только подробное описание интерфейса создаваемого учебного проекта, его функциональных возможностей, но подробно описанные пути достижения цели.

Необходимыми компонентами спецификаций являются:

- 1) проблема;
- 2) актуальность;
- 3) цели проекта;
- 4) задачи проекта;
- 5) тип проекта;
- 6) структура проекта;
- 7) методы разработки (если это необходимо для данного типа проекта);
- 8) время работы над проектом;
- 9) формы представления результатов.

В качестве примеров для анализа компонентов спецификаций, воспользуемся тремя успешно и неоднократно реализованными нами в процессе обучения информатике проектами: а) векторным графическим редактором, б) обучающей программой по одной из тем математики, в) сайтом класса (школы, группы, факультета)

1. ПРОБЛЕМА

Проблема, разрешаемая в процессе создания проекта, формулируется таким образом, чтобы школьники могли с достаточной степенью четкости представить процесс и конечный результат реализации проекта. При этом на первых этапах обучения с использованием метода проектов конечный результат должен быть представлен с максимальной четкостью. Это обусловлено низким уровнем самостоятельности школьников в ситуации, когда опыт в реализации проектов отсутствует или незначителен. На последующих этапах при приобретении знаний, умений и навыков в создании проектов, исходная проблема может формулироваться без акцентирования внимания на методах ее решения, с постепенным уменьшением детализации конечного результата.

Например, при создании векторного графического редактора детально описываются интерфейс и возможности программного продукта. Описание процесса реализации заключается в определении модели функционирования редактора, описании типов данных, переменных, основных процедур и функций. Аналогичным образом описываются проекты создания обучающей программы по теме "Функции" и сайта класса.

2. АКТУАЛЬНОСТЬ

В описании актуальности проекта необходимо:

- учитывать социальную значимость проекта;

- определять необходимость разработки собственного варианта программного обеспечения с целью лучшего ознакомления с принципами создания программ;

- опираться на обусловленную необходимость изучения нового материала и его значение в информатике.

Например:

1. При изучении графических возможностей компьютера школьники осваивают графические операторы (процедуры и функции) некоторого языка программирования. При решении задач на применение таких операторов возникает следующая проблема: для создания даже простейшего рисунка минимальной художественной значимости требуется приложить значительные усилия. Школьников утомляют рутинный расчет координат, перенос изображения с бумаги на экран компьютера, причем полученный алгоритм (программа) в подавляющем большинстве случаев имеет линейную структуру и не требует творческого подхода к своей реализации. Повторяющиеся, однообразно репродуктивные действия резко снижают познавательную активность ученика. Таким образом, с точки зрения школьника становится очевидной актуальность реализации векторного графического редактора. С точки зрения учителя актуальность данного программного продукта обуславливается необходимостью изучения графических возможностей языка, методов работы с графическими примитивами.

2. Актуальность проекта "Функции" основывается на практической значимости данного программного продукта, а также на необходимости изучения возможностей соответствующего языка программирования.

3. Актуальность проекта по созданию Internet-сайта класса определяется социальной значимостью данной разработки. Такой проект в значительной степени способствует как индивидуальному, так и групповому самовыражению школьников. Подключение школы к Интернет позволяет большому кругу людей о существовании школы и данного конкретного

класса, об их традициях и возможностях. В отсутствии сети Internet сайт помещается в локальной Intranet-сети школы, что производит сильный эффект на ближайшее окружение класса.

3. ЦЕЛИ ПРОЕКТА

Исследуя многообразие целей проекта, мы выделили основные их характеристики и разделили на две группы (по роли и по назначению в учебном процессе).

К первой группе мы отнесли следующие цели:

1. Проект как источник знаний. Эта цель наиболее актуальна в исследовательских, информационных проектах.

2. Создание проекта как компонент развития личности. К развивающимся качествам личности следует также отнести познавательный интерес, коммуникативные умения, алгоритмическое мышление и т.д.

3. Содержание обучения. Процесс работы над проектом сам является содержанием обучения и относится к разделу «Методы создания больших программных комплексов».

Во второй группе мы выделили:

1. Дидактические функции:

- обучающе-теоретические;
- обучающе-практические;
- закрепляющие;
- контролирующие.

Они отвечают за научную актуальность, методы обучения и познавательную насыщенность проекта.

2. Развивающие функции:

- познавательно-личностные;
- мировоззренческие.

Они отвечают за развивающие особенности определенного типа проекта.

3. Воспитательные функции:

- нравственная;
- эстетическая.

Комплекс целей проектов представлен на Рис. 10.



Рис. 10. Комплекс функций проектов.

В приведенных ниже примерах выделим только основные, доминирующие цели проектов:

1. Векторный графический редактор.

Образовательная цель:

- изучение основ языка программирования;
- изучение графических возможностей языка.

Воспитательная:

- воспитание эстетического вкуса.

Развивающая:

- развитие теоретического мышления;
- развитие творческих способностей;
- развитие коммуникативных умений.

2. Проект "Функции".

Образовательная цель:

- изучение основ визуальной среды программирования;
- актуализация школьного курса алгебры и начала анализа по теме "Функции".

Воспитательная:

- воспитание чувства ответственности за предлагаемый программный продукт перед его пользователями;
- воспитание эстетического вкуса в оформлении представляемого материала.

Развивающая:

- развитие умений адекватной оценки и самооценки;
- развитие внимания, творческих способностей.

3. Сайт класса.

Образовательная цель:

- изучение HTML, языка программирования JavaScript, средств создания гипертекстовых страниц и подготовки изображений;
- обучение основам работы в Internet.

Воспитательная:

- воспитание бережного отношения к информации.

Развивающая:

- развитие умений наглядно представлять информацию;
- развитие творческих способностей;
- развитие коммуникативных умений.

4. ЗАДАЧИ ПРОЕКТА

К задачам проекта отнесем:

- изучения нового материала по определенной тематике;
- создание программного комплекса;
- описание его в виде методического пособия для пользователя.

1. Выделим задачи проекта "Векторный графический редактор":

- изучение графических возможностей языка;
- создание программного продукта "Векторный графический редактор";
- создание правил пользования данным продуктом с полным описанием его интерфейса и возможностей.

2. Задачами проекта "Обучающая программа по математике" являются:

- актуализация знаний по соответствующему разделу математики, а также углубленное изучение и систематизация знаний в этом разделе. Изучение методов построения интерфейса программных продуктов;
- создание обучающей программы;
- реализация правил пользования программой.

3. К задачам проекта "Сайт класса" отнесем:

- изучение языка HTML и специализированных средств создания Web-страниц (Microsoft FrontPage, Netscape Composer и др.);

- планирование структуры и разработка сайта, представляющего собой связный гипертекстовый документ, состоящий из набора Web-страниц;

- описание правил работы с сайтом не предусматривается, так как принципы доступа к гипертекстовому документу единообразны в пределах браузера.

Одной из важнейших задач практически любого программного проекта является описание правил пользования. Сюда относится как результирующая документация по проекту, так и контекстная помощь, которая должна быть включена с программный проект отдельным компонентом. Разработка документации позволяет школьнику представить программный проект как единое целое, систематизировать его возможности, увидеть расхождения с исходной моделью. Как показывает опыт, именно на этапе разработки документации определялись концептуальные несоответствия компонентов готового проекта друг другу и основной идее в целом, что служило причиной его частичной переработки.

5. ТИП ПРОЕКТА

Типология проектов была нами проанализирована в параграфе 1.2. В рассматриваемых примерах тип графического редактора выражается формулой $M_{\text{ТВ-иссл}}C_{\text{уч-прикл}}F_{\text{кв-рут}}K_{\text{откр}}X_{\text{внутр}}Y_{\text{пар}}P_{\text{долг}}$, тип обучающей программы по математике - формулой $M_{\text{инф-иссл}}C_{\text{уч-прикл}}F_{\text{кв-рут}}K_{\text{откр}}X_{\text{внутр}}Y_{\text{инд}}P_{\text{ср}}$, а тип сайта класса – формулой $M_{\text{ТВ-инф}}C_{\text{уч-прикл}}F_{\text{рут}}K_{\text{откр}}X_{\text{внутр}}Y_{\text{груп}}P_{\text{ср}}$.

6. СТРУКТУРА ПРОЕКТА

Структура проекта описывается в виде требований к методическому пособию и к программному комплексу, представляемых пользователем. Здесь обязательно указывается все необходимые свойства проекта и его

функции, а также желательные и нежелательные аспекты использования программного продукта.

Например, в реализуемом графическом редакторе описываются виды используемых графических примитивов (линии, прямоугольники, ломаные, окружности, битовые образы и т.д.), свойства примитивов (карандаш, кисть и др.), все основные функциональные блоки, отвечающие за действия над примитивами (создание, удаление, перемещение, запись, чтение и т.д.) и принцип связи данных блоков в единое целое (например, событийная или объектно-событийная организация). Затем рассматривается интерфейс редактора (общий вид, комбинации клавиш, используемые интерфейсные блоки – меню, зависимые и независимые кнопки, переключатели, диалоговые средства и т.д.). Более подробно данная структура описана в учебном пособии [158]

В обучающей программе по математике (тема – "Функции") при описании структуры сначала раскрываются параметры исследуемого объекта, а затем – реализуемый интерфейс.

Структура сайта класса характеризуется акцентированием внимания на указании представляемой тематики и единообразии интерфейса.

7. МЕТОДЫ РАЗРАБОТКИ ПРОЕКТА

Описание методов создания программного комплекса, служит для подготовки реализации предыдущего пункта и в значительной степени зависит от возможностей используемого языка программирования. Для групповых проектов обязательно указываются функциональные обязанности каждого участника группы.

Например, в графическом редакторе описываются типы данных для хранения графических примитивов и функции их обработки, план реализации выбранной модели программирования, методы создания интерфейсных блоков. Похожим образом описываются обучающая программа по ма-

тематике. В реализации сайта класса особое внимание уделяется раскрытию методов взаимодействия членов группы, а также методам интеграции индивидуального материала в единую модель проекта.

8. ВРЕМЯ РАБОТЫ НАД ПРОЕКТОМ

Время создания проекта описывается в количественных единицах, детализируя соответствующий типологический признак проекта и варьируется от двух уроков до одного учебного года. Например, продолжительность создания графического редактора – одна-две четверти, обучающей программы по математике – от 3 до 10 занятий, сайта класса – от 2 занятий до одной четверти в зависимости от информационной насыщенности сайта.

9. ФОРМЫ ПРЕДСТАВЛЕНИЯ РЕЗУЛЬТАТА

В качестве результата выполнения проекта может быть программный продукт или комплекс программ со следующими дополнительными компонентами:

- методическое пособие для пользователя;
- статья;
- доклад, выступление на конференции;
- реферат и т.д.

Разработка спецификации проекта – это результат совместной деятельности учителя и обучаемых в общем случае. Однако в зависимости от уровня готовности учащихся к проектной деятельности смещается приоритет главенствования отдельных субъектов обучения в данном процессе. В соответствии с этим выделим три временные стадии создания спецификаций проекта:

Стадия 1: Спецификацию проекта, его основные компоненты описывает сам учитель. Он же предлагает и темы работ.

Стадия 2: Темы проектов, их цели, задачи, актуальность описывают школьники, учитель же предлагает им структуру и методы создания проекта, также он выбирает форму представления результатов проекта.

Стадия 3: Весь процесс описания спецификаций берут на себя школьники. Роль же учителя заключается в подсказке и проверке полученного. На данной стадии могут быть также следующие варианты взаимодействия учителя и школьника:

- учитель предлагает тему исследования;
- учитель только утверждает, возможно – после переформулировки, предложенную школьниками тему.

Построение спецификации программного проекта является обеспечением процесса целеполагания, то есть условием успешности четкого и однозначного выделения целей в методе проектов, а также путей их достижения. В то же время спецификация является динамичной структурой, с возможностями видоизменения в случае, когда результат реализации ее отдельного компонента вступает в противоречие с частью или со всем исходным планом. Трансформация спецификации может продолжаться вплоть до завершения учебного проекта, а полученный результат будет основным источником для формирования программной документации и оперативной помощи проекта.

2.3. Методика обучения программированию в 10-11 классах общеобразовательной школы на основе использования метода проектов

В условиях вариативности и многообразия учебных планов, учебников, школьных и авторских программ по информатике министерством образования Российской Федерации в 1999 году был утвержден обязательный минимум содержания среднего (полного) общего образования по информатике (приказ от 30.06.99 N 56) двух уровней (А и Б). Целью данной инициативы является сохранение общего ядра базового образования по информатике, т.е. формирование представлений о сущности информации и информационных процессов, развитие навыков алгоритмического мышления, являющегося необходимой частью научного взгляда на мир, знакомство учащихся с современными информационными технологиями. Требования первого уровня (А) предъявляются учащимся любого общеобразовательного учреждения вне зависимости от его типа и направления, а также наличия или отсутствия компьютерной техники. Второй уровень (Б) предназначен для учреждений, обладающих компьютерной техникой. Выполнение требований второго уровня позволяет учащимся не только получить представление об основных понятиях информатики и успешно завершить обучение, но и продолжить свое образование по данной специальности.

Специфика применения метода проектов в обучении программированию подразумевает наличие компьютерного класса, инструментария для реализации целей проекта. Таким образом, предлагаемая нами методика относится к уровню Б общеобразовательного стандарта, содержательной линии "Алгоритмизация и программирование". При этом в той или иной степени, в зависимости от тематики учебного проекта, охватываются и все остальные линии, как в актуализационном, так и в пропедевтическом плане.

Выделим необходимые предварительные условия для успешного и эффективного применения метода проектов в обучении программированию:

- 1) наличие компьютерного класса, с возможностью доступа к нему школьников во внеурочное время;
- 2) наличие компилятора или интерпретатора языка программирования, с применением которого изучается учебный материал соответствующего раздела информатики;
- 3) наличие достаточной компетентности учителя, что подразумевает его предварительное участие в разработке больших программных проектов (не менее 2-3 проектов, разработанных самостоятельно или в группе).

Невыполнение любого из этих условий приводит к нивелированию положительного эффекта применения метода проектов по сравнению с традиционными методиками обучения программированию.

Придерживаясь технологической схемы метода проектов, представленной в параграфе 1.3, проанализируем методику организации процесса обучения программированию с учетом уровней готовности школьников к реализации учебного проекта.

Для этой цели выделим три соответствующих уровня готовности – *начальный, средний и продвинутый*. Такое разделение подразумевает понимание школьниками основных принципов метода проектов, наличие опыта в реализации учебных проектов и. Ниже приводится набор знаний, умений и навыков, согласно которым определяется принадлежность обучаемых к одному из уровней готовности (Таблица 5). Следует учесть, что знания в области собственно программирования не оказывают решающего влияния на уровень готовности к проектной деятельности. Они имеют значение только для степени сложности проекта с точки зрения использования специальных приемов программирования.

Таблица 5

Уровни готовности школьников к осуществлению проектной деятельности

	Начальный (нулевой) уровень	Средний уровень	Продвинутый уровень
Знания основ проектной деятельности	Обучаемый планирует и осуществляет работу над проектом методом проб и ошибок	Обучаемый знает основные этапы работы над проектом, может прогнозировать их реализацию	Обучаемый знает основы проектной деятельности и с успехом может применить их на практике
Умение составлять спецификацию проекта	<p>На этом уровне разработка спецификации ведется в основном учителем.</p> <p>Учащиеся могут самостоятельно составить спецификацию только в самом общем виде (придумать название проекта, ограничить временные рамки, обозначить результат). При этом в спецификации они учитывают только результативный, но не процессуальный аспект.</p>	<p>Учащиеся разрабатывают спецификации проектов под руководством учителя. Самостоятельное создание спецификации может иметь место при наличии предварительного подробного анализа создаваемого проекта, выполненном учителем.</p> <p>В спецификации могут быть не учтены некоторые аспекты результативного характера, а процессуальный аспект проекта рассматривается только в самом общем виде</p>	<p>Разработка спецификаций ведется как самостоятельно, так и с помощью учителя. Школьники представляют проект как в общем виде, так с точки зрения реализации отдельных блоков. В спецификации учитываются как результативный, так и процессуальный аспекты разрабатываемого проекта</p>

<p>Умение планировать деятельность по созданию проекта с известной спецификацией</p>	<p>Учащиеся владеют навыками решения задач по программированию, что позволяет им осуществлять проектную деятельность на этапе создания отдельных программных блоков. При этом они не видят общей структуры проекта, в результате чего их решения с трудом состыкуются друг с другом. Процесс объединения программных блоков в единое целое сталкивается со значительными трудностями</p>	<p>Учащиеся владеют навыками программирования отдельных блоков проекта. Они не только решают поставленные задачи, но и пытаются придерживаться некоторой схемой решения, выделяют общие композиционные исходные данные, определяют для каждой подзадачи входные и выходные параметры</p>	<p>Учащиеся строят общую схему решения и последовательно ее придерживаются. При решении отдельных блоков учащиеся стремятся к максимальной универсализации программного кода, который впоследствии можно будет использовать при реализации других проектов.</p>
<p>Умение планировать деятельность по созданию проекта с известной спецификацией</p>	<p>Планирование проектной деятельности осуществляется только под руководством учителя.</p>	<p>Учащиеся могут по готовой спецификации разработать план работы, разбить проблему над подзадачами, распределить роли в группе. Однако для подзадач не выделяются критические параметры, не выделяются четко входные и выходные данные.</p>	<p>Учащиеся данного уровня умеют планировать проектную деятельность как для своей, так и для других групп. Четко выделяют необходимые компоненты данной деятельности, разбирают проблему над подзадачами и строго определяет входные и выходные параметры. Однозначно определяют для себя конечный результат</p>

Для иллюстрации выдвинутых положений воспользуемся учебным программным проектом "Векторный графический редактор" (ВГР), представляющим собой интерактивную инструментальную среду, создание которой является нетривиальной задачей даже для профессионального программиста. Сложность проектирования, большое количество отдельных программных блоков с многоуровневыми связями, длительность разработки ограничивают применимость данного проекта классами с углубленным изучением информатики. Однако в силу тех же причин в процессе разработки ВГР находят свое отражение практически все аспекты предлагаемой нами методики применения метода проектов в обучении программированию.

Рассмотрим результативный аспект спецификации ВГР в максимально сокращенном виде:

1. Графический редактор позволяет манипулировать простейшими графическими примитивами, как то – отрезками, окружностями, прямоугольниками, эллипсами, многоугольниками, заливками различных видов и т.д. Под манипуляциями понимаются интерактивное создание, удаление, изменение местоположения, размеров и других атрибутов графических примитивов.

2. Редактор позволяет хранить (запись и чтение) на диске полученный набор примитивов в специализированном формате, а также сохранять в текстовом файле набор вызовов графических процедур и функций. Таким образом, подготовленный в редакторе рисунок можно затем использовать в любой программе на языке программирования Турбо-Паскаль.

3. Редактор имеет достаточный для выполнения своих функций набор элементов управления – зависимые и независимые кнопки, поля для выбора цвета и других атрибутов графических примитивов, поля ввода и вывода текста, средства скроллинга, меню и т.д.

4. Основное средство управления редактора является манипулятор "мышь". Для ввода текстовой информации и управления с помощью "горячих клавиш" используется клавиатура.

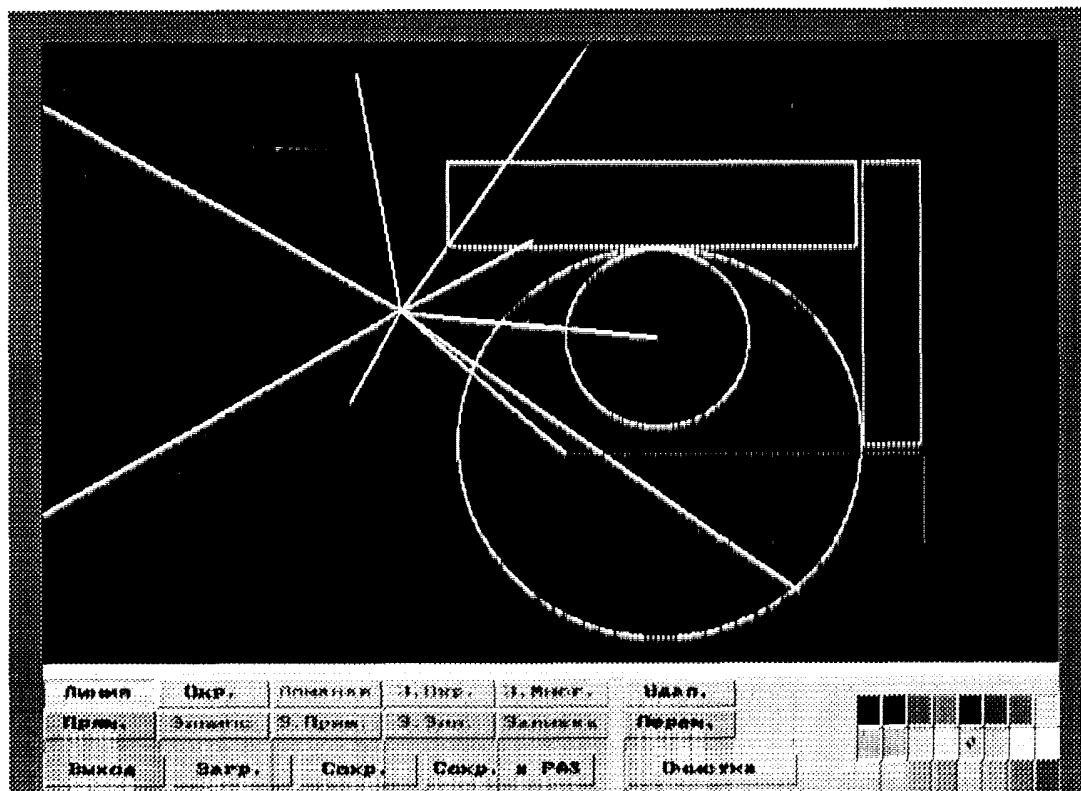


Рис. 11. Примерный вид векторного графического редактора.

Согласно технологической схеме раскроем особенности методики применения метода проектов при обучении программированию

ПЕРВЫЙ (ИСХОДНО-УСТАНОВОЧНЫЙ) ЭТАП ТЕХНОЛОГИЧЕСКОЙ СХЕМЫ МЕТОДА ПРОЕКТОВ:

НАЧАЛЬНЫЙ УРОВЕНЬ ГОТОВНОСТИ

(Исходно-установочный этап).

Учитель делает краткий обзор изучаемого материала, формулирует и актуализирует возникающие проблемы, предлагает школьникам один или

нескольких программных проектов, разработка и реализация которых решает эти проблемы.

Совместно с учителем ученики производят выбор проекта, взвешивая все "за" и "против". Обязательным условием успешности данного этапа является осознание школьниками необходимости создания именно этого программного продукта, его уникальности, полезности и прагматичности.

Затем ученикам предлагается готовая спецификация учебного проекта, причем учитель логическими, а не авторитарными методами добивается понимания и осознания необходимости каждого из разделов спецификации. Одним из важнейших условий эффективности процесса обучения с точки зрения адекватности набора получаемых школьником знаний умений и навыков с государственным стандартом является прямой или косвенный охват программным проектом всего теоретического материала, изучаемого на протяжении реализации проекта. При этом не допускается уменьшение объема фактологического материала, разрешается только изменение последовательности изложения слабосвязанных друг с другом блоков и увеличение объема специфичной для метода проектов информации.

Например, при изучении графических операторов языка программирования возникает следующая проблема – продолжительность и трудоемкость создания заданного рисунка неадекватна получаемому результату. Это относится прежде всего к рисункам художественной направленности, в которых нельзя выделить алгоритмизируемую последовательность действий, статическим элементам интерфейса и оформления. Расчет координат, выбор цветов, стилей, средств рисования является трудоемким, утомительным процессом и формирует только репродуктивные умения и навыки. Для решения этой проблемы учитель предлагает разработать учебный векторный графический редактор, с возможностями интерактивного создания произвольных векторных изображений и сохранения подготовлен-

ного рисунка в командах изучаемого языка. Затем учитель предлагает спецификацию программного проекта, включающую требования к интерфейсу, возможностям, а также ограничения, налагаемые на проект особенностями языка программирования, операционной системой и изучаемым материалом.

СРЕДНИЙ УРОВЕНЬ ГОТОВНОСТИ

(Исходно-установочный этап).

В отличие от начального уровня, на данном этапе формулировка и актуализация проблемы производится совместными действиями учителя и ученика. Аналогичным образом строится спецификация программного проекта. Авторитарная роль учителя проявляется, во первых, в очерчивании круга решаемых проблем, во вторых – в построении спецификации или в выборе одного из предложенных учениками вариантов спецификации. В последнем случае учитель обязан "защитить" свой выбор цепочкой логических рассуждений, причем мотивация вида "Данный вариант - наилучший, так как он иллюстрирует все темы, которые мы будем изучать в этой четверти" недопустима. Причины выбора той или иной спецификации могут опираться только на глобальную цель – создание программного проекта соответствующей тематики.

Например, предложение построения графических примитивов в виде объектов (язык Turbo Pascal) не может мотивироваться необходимостью иллюстрации принципов объектно-ориентированного программирования. В данном случае следует ссылаться, например, на требование унификации манипуляций с примитивами. Так как школьникам в данный момент еще неизвестно понятие "объект", можно дать его пропедевтическое определение совместно понятиями инкапсуляции и наследования, а также чисто интуитивное понятие полиморфизма. Глубина рассмотренных понятий должна быть достаточна для доказательного выбора требуемой альтернативы спецификации.

ПРОДВИНУТЫЙ УРОВЕНЬ ГОТОВНОСТИ

(Исходно-установочный этап).

В отличие от среднего уровня, поиск проблемы на основе фактов, сообщаемых учителем, ученики производят самостоятельно. Аналогичным образом производится поиск тематики программного проекта и создание спецификации. Учитель выступает в роли аналитика-консультанта, мнение которого является важным, но не решающим. Авторитарный подход со стороны учителя может использоваться только в случае, если выбранный учениками программный проект не вписывается во временные рамки, отведенные для его реализации. На этом уровне школьники уже понимают, что весь изучаемый материал должен найти свое отражение в проекте, поэтому учитель может делать предложения, недопустимые по своей мотивации на начальном и продвинутом уровнях.

Например, выбрав программным проектом векторный графический редактор ученики самостоятельно готовят его спецификацию, разбивают проект на подзадачи, распределяют роли и т.д. Учитель помогает, консультирует, предлагает использовать те или иные специфические средства для выполнения отдельных задач (поток – для хранения примитивов на диске, объектно-событийную модель – для организации ввода-вывода и реакции программы на события и т.д.), причем принципы организации таких средств ученикам могут быть заранее неизвестны.

ВТОРОЙ (ИСХОДНО-ДИАГНОСТИЧЕСКИЙ) ЭТАП ТЕХНОЛОГИЧЕСКОЙ СХЕМЫ МЕТОДА ПРОЕКТОВ.

НАЧАЛЬНЫЙ УРОВЕНЬ ГОТОВНОСТИ

(исходно-диагностический этап).

На этом этапе учитель определяет текущий уровень и актуализирует знания учащихся, мотивируя необходимостью их применения для создания программного проекта. При этом положительным мотивационным эффек-

том будет обладать указанием конкретных блоков проекта, реализация которых невозможна без соответствующих знаний в области языка или методов программирования.

Например, актуализируя понятие цикла (операторов, организующих цикл) следует показать, что данное понятие активно будет использоваться в графическом редакторе, например, при организации получения команд с устройств ввода, перебора графических примитивов и т.д.

СРЕДНИЙ УРОВЕНЬ ГОТОВНОСТИ

(исходно-диагностический этап).

Мотивация актуализации имеющихся у школьников знаний должна быть более развернутой, чем на начальном уровне, так как по своему опыту в создании проектов учащиеся понимают, что требуемые для реализации проекта знания являются небольшим подмножеством знаний по программированию. Учитель может опираться только на утвержденную ранее спецификацию проекта.

Например, актуализируя понятие записи (язык Turbo Pascal), учитель не может опираться на рассуждение вида "Мы будем изучать графику, где активно используются записи для описания различных структур". Его рассуждения могут иметь вид "Графические примитивы мы будем реализовывать (см. спецификацию) с помощью объектов, которые является логическим развитием понятия запись. Следовательно, мы должны вспомнить все, что нам известно о записях"

ПРОДВИНУТЫЙ УРОВЕНЬ ГОТОВНОСТИ

(исходно-диагностический этап).

На этом этапе самостоятельность школьников выражается в выборе материала для повторения. Учитель имеет право расширить список актуализируемых тем.

Например, учащиеся могут выбрать для повторения материалы, касающиеся взаимодействия с манипулятором "мышь", посчитав ненужным

информацию о вводе с клавиатуры. Учитель должен исправить это упущение, напомнив школьникам, что, несмотря на обеспечение подавляющего большинства операций ввода "мышью", имя файла, например, для записи примитивов на диск им придется вводить с клавиатуры.

ТРЕТИЙ (СОДЕРЖАТЕЛЬНО-ПЛАНОВЫЙ) ЭТАП ТЕХНОЛОГИЧЕСКОЙ СХЕМЫ МЕТОДА ПРОЕКТОВ.

НАЧАЛЬНЫЙ УРОВЕНЬ ГОТОВНОСТИ

(содержательно-плановый этап).

Начиная с данного этапа учитель производит процесс обучения, конкретизируя передаваемые знания в элементах спецификации программного проекта. Учитель излагает новый материал, опираясь на разработанную ранее структуру спецификации, актуализируя получаемые учеником знания в отдельных компонентах проекта. По окончании этапа основные, наиболее крупные блоки спецификации будут полностью опираться на знания школьников. Однако получение новых знаний на последующих этапах может привести к переработке исходной схемы и, соответственно, к возврату на данный этап, что является прямым следствием цикличности процесса реализации проекта.

Например, при реализации ВГР один из блоков спецификации определяет, что интерактивные манипуляции в редакторе ведутся над графическими примитивами. Однако типы примитивов и их свойства не могут быть однозначно определены без знаний о соответствующих графических операторах (функциях, процедурах) конкретного языка программирования. Таким образом, по мере изучения графики учитель постепенно наполняет данный блок спецификации конкретным содержимым.

СРЕДНИЙ УРОВЕНЬ ГОТОВНОСТИ

(содержательно-плановый этап).

На данном уровне учащиеся обладают высоким уровнем самостоятельности в конкретизации основных блоков спецификации. Учитель акцентирует внимание на тех разделах нового материала, которые необходимы для уточнения спецификации. Ученики самостоятельно определяют требуемые типы данных, процедуры и функции, программные методы последующей реализации блоков спецификации. Учитель контролирует обучаемых, предлагает произвести выбор из нескольких разделов получаемых знаний наиболее оптимальные для достижения требуемого результата, а именно – полной конкретизации спецификации программного проекта.

Например, при конкретизации блока спецификации ВГР, отвечающего за набор графических примитивов, учитель не описывает типы примитивов в терминах языка программирования. Совместно с учащимися он только перечисляет в процессе изучения материала требуемые примитивы, в то время, как описание их свойств и методов обработки учащиеся должны освоить самостоятельно. Учитель контролирует этот процесс, указывает на недостатки описания, недостающие или избыточные элементы создаваемых примитивов, мотивируя это причинами, прямо или косвенно связанными с результатом проектной деятельности.

ПРОДВИНУТЫЙ УРОВЕНЬ ГОТОВНОСТИ

(содержательно-плановый этап).

Ярко выраженная самостоятельность, знания и умения в проектной деятельности позволяют учащимся данного уровня вести процесс конкретизации спецификации активными методами. Знания, получаемые с помощью учителя, самостоятельно перерабатываются и аккомодируются в готовые мыслительные схемы, построенные на предыдущих этапах и связанные со структурой спецификации проекта. Учитель контролирует полу-

ченный результат, вмешиваясь в процесс деятельности только по желанию учащихся, советуя и консультируя.

Например, уточняя спецификацию ВГР, учащиеся в процессе получения новых знаний самостоятельно определяют требуемые типы графических примитивов, совместно со всеми сопутствующими программными элементами. Учитель дает советы, вмешиваясь в процесс только в случае его неадекватности конечной цели – реализации программного проекта, а также контролирует полученный результат.

ЧЕТВЕРТЫЙ (СОДЕРЖАТЕЛЬНО-ТЕХНОЛОГИЧЕСКИЙ) ЭТАП ТЕХНОЛОГИЧЕСКОЙ СХЕМЫ МЕТОДА ПРОЕКТОВ.

НАЧАЛЬНЫЙ УРОВЕНЬ ГОТОВНОСТИ

(содержательно-технологический этап).

На этом этапе школьники начинают создание проекта методом "снизу-вверх". Реализуются низовые структуры, требуемые по спецификации функции. Организуется творческий поиск учащихся, создаются условия для получения школьниками знаний индуктивного характера, элементарных умений и навыков. Теоретический материал, предоставляемый учителем, заполняет программные ниши, что дает возможность более эффективной разработки элементов проекта. По окончании каждого логически завершенного блока происходит обращение к следующему (пятому) этапу технологической схемы. Реализация достаточного количества низкоуровневых блоков заканчивается переходом к концепции разработки программ "сверху-вниз", связыванием отдельных блоков в единое целое. По окончании данного процесса происходит переход к шестому этапу.

Например, при реализации структуры, содержащей набор графических примитивов, учащиеся сталкиваются с проблемой повышения ее динамичности. Для такой структуры желательна простота вставки, удаления, перемещения элементов. Организуя ее с помощью массива, школьники

вынуждены создавать сложные и малоэффективные процедуры и функции его обработки. Если в это время учитель объясняет, например, основы объектно-ориентированного программирования (в Turbo Pascal), то понятие коллекции решает вышеуказанную проблему. Аналогичным образом данную проблему решает понятие списка, изучаемое при рассмотрении динамических структур данных. Основной задачей учителя при этом является согласование реализации рассматриваемого блока проекта с изучением нового материала. Опыт показывает, что наибольший эффект в усвоении получаемых знаний получается при некотором "запаздывании" изложения нового материала по отношению к разработке соответствующего блока проекта.

СРЕДНИЙ УРОВЕНЬ ГОТОВНОСТИ

(содержательно-технологический этап).

В отличие от начального уровня, на данном этапе школьники обладают большей самостоятельностью. Учитель производит укрупнение логически завершенных блоков, требующих контроля и оценки. Учащиеся более внимательны в изучении и анализе теоретического материала, активны в самостоятельном поиске знаний. Опыт в создании программных проектов позволяет им вычленять из информации, сообщаемой учителем, элементы, необходимые для достижения основной цели.

Например, при изучении любой темы в графике, учитель уже не акцентирует внимание учеников на применимости данного материала в том или ином блоке программного проекта. Школьники сами определяют требуемый блок на основе информации о полезности излагаемых фактов.

ПРОДВИНУТЫЙ УРОВЕНЬ ГОТОВНОСТИ

(содержательно-технологический этап).

Особенности продвинутого уровня предполагают самый высокий уровень самостоятельности школьников при реализации проекта. Ученики самостоятельно вычленяют и затем расширяют требуемые им знания из тео-

ретического материала, предоставляемого учителем. Учитель наблюдает за реализацией проекта, изредка внося свои коррективы.

Например, при изучении ООП, ученики сами предлагают учителю использовать потоки и ресурсы для хранения как отдельных рисунков на диске, так и целых библиотек рисунков, просят провести иллюстрацию теоретического материала на соответствующих блоках программного проекта.

ПЯТЫЙ (ПРОМЕЖУТОЧНО-КОРРЕКЦИОННЫЙ) ЭТАП ТЕХНОЛОГИЧЕСКОЙ СХЕМЫ МЕТОДА ПРОЕКТОВ.

НАЧАЛЬНЫЙ УРОВЕНЬ ГОТОВНОСТИ

(промежуточно-коррекционный этап).

Данный этап является чрезвычайно важным для организации контроля над уровнем и процессом усвоения знаний, получения умений и навыков школьниками. Каждый логически завершённый блок проверяется учителем с помощью особых тестовых заданий, предназначенных для тестирования правильности и эффективности разработанного учащимися блока. Результат выполнения таких заданий строго индивидуален для каждого школьника и напрямую зависит от индивидуальности решения проверяемого блока. Если некоторый блок, по результатам оценки, не соответствует спецификации, то происходит возврат на четвёртый этап и доработка блока.

Например, каждый графический примитив, созданный школьником, проверяется учителем с помощью специализированного задания, в результате решения которого учеником должна быть произведена демонстрация возможностей примитива – показ, скрывание, изменение местоположения, цвета, размеров примитива и т.д. Учитель не зачитывает примитив, пока все его возможности не будут визуализированы.

СРЕДНИЙ УРОВЕНЬ ГОТОВНОСТИ

(промежуточно-коррекционный этап).

Высокий уровень самостоятельности школьников позволяет укрупнять тест-задания, превращая их в мини-проекты.

Например, в отличие от начального уровня, контроль осуществляется над целым набором графических примитивов, связанных по некоторому общему признаку. Так, скажем, тест может быть проведен одновременно для всех примитивов, поддерживающих все свойства "карандаша" – цвет, толщину, стиль, режим.

ПРОДВИНУТЫЙ УРОВЕНЬ ГОТОВНОСТИ

(промежуточно-коррекционный этап).

Особенностью данного этапа является большой объем тестируемого блока.

Например, тест-программа может охватывать в себе все графические примитивы, тест всего файлового ввода-вывода, тест всего набора действий над примитивами и т.д.

Еще одной особенностью этого и последующего этапа на максимальном уровне является достаточно большая вероятность изменения спецификации программного проекта. Тематика программного проекта была выбрана школьниками самостоятельно, в результате чего серьезные ошибки или недочеты в планировании могут быть незамечены учителем. Следствием обычно является несоответствие полученного блока проекта исходной модели, что влечет за собой ее переработку.

Например, описанная в предыдущем этапе ситуация может явиться причиной переработки спецификации. Если изначально предусматривалось только хранение отдельных рисунков в файле, то с появлением новой информации о ресурсах вполне логичным является расширение спецификации возможностью поддержки библиотек рисунков.

ШЕСТОЙ (ИТОГОВО-ОЦЕНОЧНЫЙ) ЭТАП ТЕХНОЛОГИЧЕСКОЙ СХЕМЫ МЕТОДА ПРОЕКТОВ.

НАЧАЛЬНЫЙ УРОВЕНЬ ГОТОВНОСТИ

(итогово-оценочный этап).

Последний технологический этап предназначен, во первых, для оценки соответствия созданного проекта исходной спецификации, а во вторых - для тестирования общего уровня усвоения знаний школьником, полученных умений и навыков. На данном этапе школьник обязан осуществить "защиту" своего проекта, продемонстрировать его возможности и быть готовым объяснить любой участок программного кода проекта.

При защите графического редактора, школьник обязан быть готовым ответить, например, на следующие вопросы: "Каким образом происходит сохранение рисунка в файле?", "Как реализовано перемещение графического примитива по экрану?", "Как обрабатываются события от манипулятора мышь?" и т.д. Во всех случаях требуется показать и объяснить соответствующий участок кода.

СРЕДНИЙ УРОВЕНЬ ГОТОВНОСТИ

(итогово-оценочный этап).

На данном этапе проект оценивается более строго, чем на начальном уровне. Кроме стандартных, предусмотренных спецификацией возможностей, анализируются и дополнительные, введенные школьником самостоятельно. Приветствуются любые изменения, повышающие функциональность исходной модели, либо упрощающие ее без нарушения функциональности.

ПРОДВИНУТЫЙ УРОВЕНЬ ГОТОВНОСТИ

(итогово-оценочный этап).

Следствием максимальной самостоятельности на данном уровне является максимальный набор требований к программному проекту. Во первых, каждый представляемый проект строго индивидуален в пределах

группы разработчиков. Двух похожих по реализации проектов быть не может, плагиат недопустим ни в какой степени. Во вторых, обязателен выход за пределы спецификации. Таким образом набор функций проекта расширен по сравнению с исходным планом.

Например, при сдаче двумя школьниками (из различных групп разработчиков) близких по реализации проектов, зачитывается только первый, второй направляется на переработку.

Предлагаемая нами методика обучения программированию на основе следования технологической схеме метода проектов с учетом разделения учащихся по уровням готовности к использованию метода проектов успешно прошла апробацию в школах г.Шадринска, что подтверждено результатами педагогического эксперимента.

2.4. Подготовка будущих учителей информатики к использованию метода проектов

Использование метода проектов в обучении программированию в школьном курсе информатики накладывает особые требования к подготовке будущего учителя.

Для эффективной помощи и поддержки осуществления проектной деятельности учащихся учителю необходимы следующие знания:

- ✓ знания языков программирования высокого уровня;
- ✓ знания особенностей построения спецификаций проектов;
- ✓ знания основ проектной деятельности;
- ✓ знания методов контроля за процессом создания проекта;
- ✓ знания особенностей методики обучения информатике с использованием метода проектов.

Для формирования этих знаний необходимо проанализировать и видоизменить не только программу курса МПИ педагогических вузов, готовящих специалистов по квалификации "учитель информатики", но и всего курса информатики в целом. В основных дисциплинах предметного блока необходимо выделить компоненты, которые в дальнейшем используются при создании проектов или обучении по методу проектов. Выделим эти компоненты и формируемые в рамках дисциплины знания, умения и навыки (Таблица 6).

Исходя из этой таблицы можно выделить следующие этапы формирования у будущих учителей информатики умений осуществлять процесс обучения на основе использования метода проектов:

1 этап: Формирование знаний и умений, необходимых для реализации программных проектов.

Таблица 6

Знания, умения и навыки, необходимые будущему учителю
информатики для использования метода проектов в своей
профессиональной деятельности

Дисциплина	Курс	Формируемые знания, умения и навыки
Программирование	1-2	<ul style="list-style-type: none"> ✓ знания языков программирования; ✓ умения самому создавать проект; ✓ навыки составления отдельных процедурных блоков проектов; ✓ умения работать в группе программистов, распределять роли, руководить людьми; ✓ знания методов контроля за текущим состоянием работы и конечным результатом
ПО ЭВМ	1	<ul style="list-style-type: none"> ✓ знания основ проектной деятельности; ✓ знания прикладной направленности метода проектов; ✓ умения по четкой спецификации определять необходимые структурные и композиционные компоненты проекта; ✓ умения правильно оформлять необходимую документацию к проекту
ПРЗ на ЭВМ	1-2	<ul style="list-style-type: none"> ✓ умения решать небольшие задачи; ✓ умения видеть необходимые исходные и выходные данные для той или иной функции или процедуры; ✓ знания основных нечисловых и численных алгоритмов (в том числе и алгоритмов поиска и сортировки); ✓ умения оптимизировать алгоритм исходя из конкретных исходных данных
ЧМ	2	<ul style="list-style-type: none"> ✓ знание основных численных алгоритмов, используемых при разработке проектов, требующих сложных математических расчетов (например, связанных с построением фигур, движущихся объектов, с построением модели явления, основанном на решении системы линейных уравнений или определенных интегралов и т.д.)

INTERNET и компьютерные сети	4	<ul style="list-style-type: none"> ✓ знания основ использования ресурсов INTERNET при реализации метода проектов как в информатике, так и в других дисциплинах школьного курса; ✓ умения работы в сети INTERNET; ✓ знания основных методов работы со всемирной сетью
МПИ	3-4	<ul style="list-style-type: none"> ✓ знания основ проектной деятельности; ✓ знания основных особенностей построения методики обучения информатики с использованием метода проектов; ✓ умения планировать процесс обучения школьников информатике на основе использования метода проектов; ✓ знания возрастных особенностей школьников при работе над проектом; ✓ умения планировать работу учащихся по созданию проекта;
		<ul style="list-style-type: none"> ✓ умения составления спецификаций проектов с учетом уровней сформированности проектных умений у школьников, их возраста и перспективных возможностей языка программирования; знание технологической схемы написания учебного проекта учащимися среднего и старшего звена общеобразовательной школы

Этап захватывает 1-2 курсы обучения и реализуется на основе следующих дисциплин: ПО ЭВМ, ПРЗ на ЭВМ, ЧМ, программирование. Выделение этого этапа как основополагающего для дальнейшего обучения соответствующей методике опирается на следующие объективные причины:

- ✓ отсутствие у некоторых студентов опыта проектной деятельности как такового;

- ✓ отсутствие у некоторых студентов знаний основ языков программирования (это связано, в первую очередь, с разрешением принятия вступительного экзамена по информатике в ВУЗ только на альтернативной основе, т.е. по желанию абитуриента. Поэтому среди студентов встречаются

те, кто вообще информатику не изучал или получил в школе только пользовательские навыки);

✓ отсутствие опыта создания сложных программных комплексов, требующих дополнительных знаний по возможностям языков высокого уровня и т.д.

Обучение программированию на основе использования метода проектов должно быть осуществлено на основе создании больших программных комплексов. Для этого требуется работать в группе программистов, что сразу ставит студентов в условия необходимости руководства или подчинения некоторым членам группы.

Роли в группе могут быть распределены следующим образом:

1. Главный программист. Основные обязанности: пишет и отлаживает программу, согласовывает входные и выходные спецификации. Наиболее квалифицированное звено группы.

2. Помощник главного программиста. Может взять на себя выполнение любой части поставленной задачи. До тонкостей знает всю программу. Ищет альтернативные стратегии решения.

3. Инструментальщик. Разрабатывает ресурсозависимые приложения программы.

4. Контролер. Готовит тесты для программы и прогоняет их на каждом варианте решения задачи. Отвечает за констатацию ошибок.

5. Редактор. Отвечает за правильное оформление всей документации по проекту.

6. Программисты. Бригады программистов, занимающиеся разработкой отдельных блоков программного продукта. [30]

В целях наиболее эффективной работы над проектами необходимо, чтобы за время обучения студент поработал в каждой из представленных должностей.

Процесс работы над общей структурой проекта, особенно при работе со студентами, первый раз сталкивающимися с данным методом, процесс обучения можно представить в виде некоторой последовательности действий строго описанной в соответствии с технологической схемой и основными правилами групповой работы.

Выделим эти правила:

1. Малая группа может быть выбрана на любом основании (игровая жеребьевка, желание обучаемых и т.д.).
2. Предпочтительны "смешанные" группы, т.е. нежелательно образование групп только из «сильных» или только из «слабых» студентов.
3. Любое высказывание должно быть аргументированным, безапелляционные высказывания не допустимы.
4. Аргументы опираются на знания, факты и опыт, а не на силу, мнения авторитетов, устоявшиеся догмы.
5. Эффект деятельности группы значительно возрастет, если защищаемый проект «проходит» не из-за собственной ценности, реальной или предполагаемой, а благодаря искусству защищающей его группы, так как и в реальной жизни принимаются лучше аргументированные предложения.
6. Тон высказывания уважительный. Обязательно обращение к оппонентам по именам и «озвучивание» их позиции.
7. По окончании дискуссии не обязательно из нескольких вариантов выбирать один – лучший, напротив, учащийся должен привыкать к многоцветности истин, естественности несовпадающих путей к одному и тому же результату.

Наиболее часто используются следующие ситуации для дискуссии:

1. Ситуация «Защита темы» – ставила субъектов обучения в условия, когда результат деятельности группы зависит от умения актуализировать значимость изучаемого, ориентации учащихся в вопросах познания в области программирования, умения работать в пространстве диалога, отби-

рать и транслировать учебный материал с точки зрения его перспективных возможностей. С позиции метода проектов эта ситуация рассматривается как возможность защиты гипотезы о наиболее эффективных путях создания проекта.

2. Ситуация «Собственная позиция» ставила субъектов перед необходимостью решения различных профессиональных задач, в которых актуализировалось выражение собственного мнения по поводу содержания теоретических категорий и понятий, ценностный выбор, самообоснование своих действий в рамках метода проектов.

3. Назначение ситуации «Принятие решения» состояло в разрешении задач, связанных с построением проекта, осознание ответственности за принимаемые решения в условиях, в которых собственные достижения и самореализация собственной личности зависит от умения принимать решения, приводящие к самореализации других субъектов ситуации, что связано с умением распределять роли в разработке проекта, делегировать функции, стимулировать потребности в достижении, преодолении трудностей, актуализирующих саморазвивающую деятельность субъектов образовательного процесса.

4. Технологическим инструментарием создания ситуации «самоотрицание» была постановка учащегося в условия, когда эффективность деятельности предполагала активную рефлексию собственного опыта на основе специально разработанных заданий, программ и тестов.

В работе над проектами мы активно использовали такую форму организации процесса обучения, как проблемный семинар, разбитый на стадии следующим образом:

1. Постановка и осмысление проблемы.

Преподаватель или учащийся излагает состояние рассматриваемого вопроса, уточняя имеющиеся представления о его сущности. Затем описывается или намечается то, что требуется получить. В ходе краткой (не бо-

лее 5 минут) беседы студенты уточняют свое видение более детально, хотя и не окончательно, прорисовывают смысл противоречий. Можно считать, что проблема осмыслена, когда она вербализирована, т.е. сформулирована в виде задачи.

2. Генерирование вариантов решения проблемы.

Типовая процедура «Мозговой штурм».

3. Поиск аргументов в поддержку предложенных решений.

Аудитория делится при этом на группы, по числу зафиксированных предложений. Представитель группы («спикер») наугад вытягивает карточку с номером предложений. Задача группы – за 8-10 минут найти как можно больше аргументов в пользу доставшихся им вариантов решения, даже если эти варианты не очень им нравятся.

4. Отбор наиболее аргументированных вариантов решения.

Представитель каждой группы получает 1-2 минуты для изложения своих аргументов. После того, как будут прослушаны все сообщения, начинается обсуждение, его цель – отобрать приблизительно половину предложенных вариантов для дальнейшей проработки.

5. Критика отобранных решений.

Из карточек с номерами изымаются те, что не прошли предыдущий этап. Аудитория вновь делится по числу оставшихся вариантов, при этом сами группы численно увеличиваются. Представители групп вновь вытягивают номера из оставшихся предложений. Задача группы на ближайшие 5-8 минут – острая, но доказательная критика оставшихся вариантов.

6. Отбор решений наиболее устойчивых к критике.

Аналогично пункту 4. Особо отметим как предмет заботы преподавателя позитивный характер и этого обсуждения: требуется не отбраковывать слабые решения, а отбирать сильные.

7. Продумывание способов реализации решений.

Из карточек снова изымаются номера, не прошедшие предыдущий этап. Аудитория вновь делится на группы (большие) по числу 2-3 оставшихся вариантов. Эти большие команды при необходимости делясь на подгруппы, должны разработать конкретные способы реализации оставшихся предложений.

8. Обсуждение способов реализации решений.

На данном этапе допускается как позитивные, так и негативные выступления, причем предпочтение отдается конструктивным суждениям.

9. Подведение итогов.

В этот этап входит ряд процедур:

1) характеристика преподавателем деятельности участников и ее результатов, нацеленная на:

- вскрытие дополнительных ходов и возможностей, оставшихся вне обсуждения;
- незамеченные студентами детали и тонкости;
- существующие и возможные связи обсуждавшегося материала с содержанием других тем курса или других предметов;

2) планирование конкретных действий;

3) рефлексия участников дискуссии (лучше групповая, что требует еще одного деления на малые группы).

Полученная организационная модель приведена ниже (Рис. 12).

По приобретении студентом опыта в реализации трех – четырех проектов, можно приступить к преподаванию методики обучения информатике (раздел – алгоритмизация и программирование) на основе использования метода проектов.

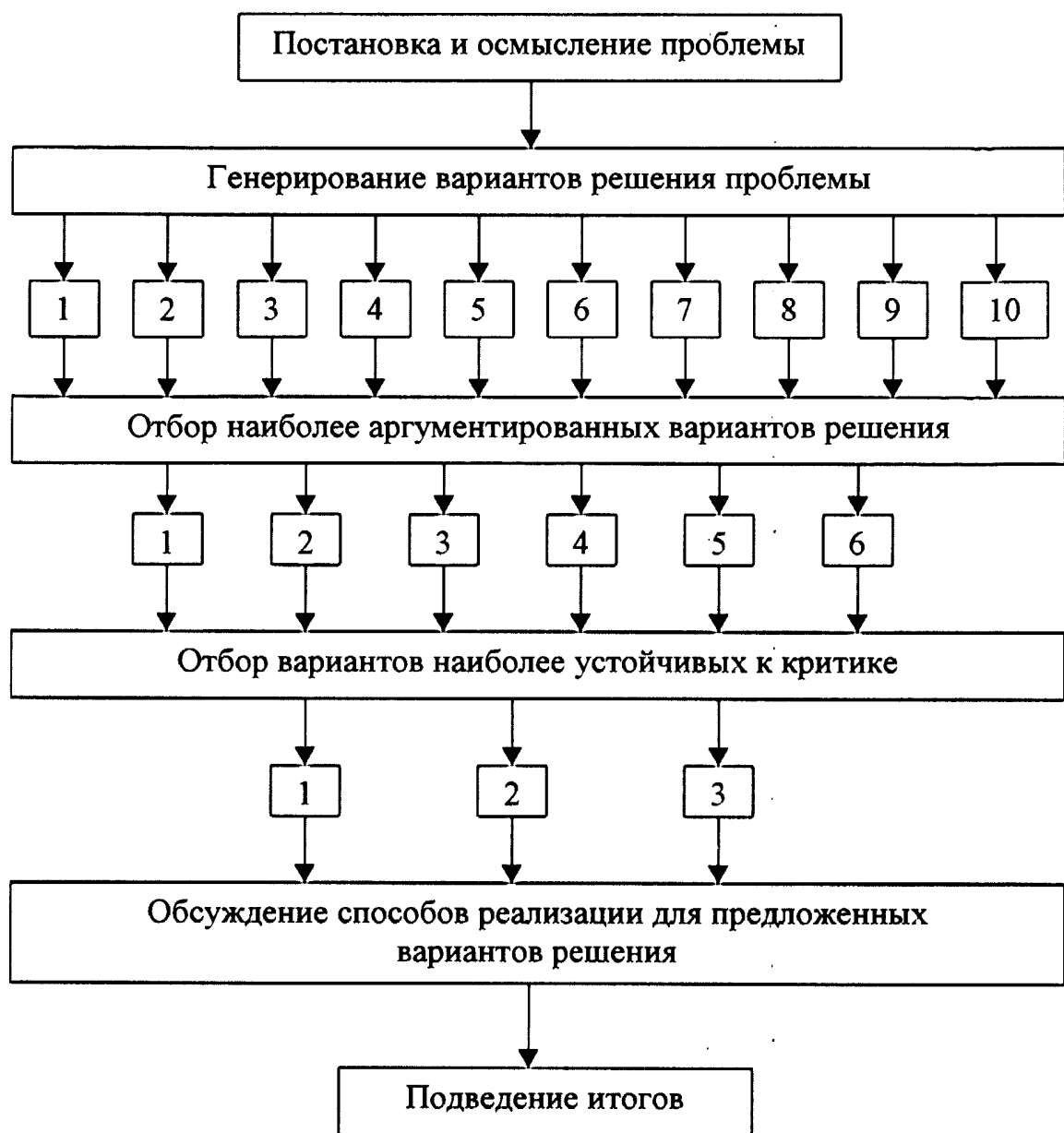


Рис. 12. Организационная схема проблемного семинара

2 этап: Обучение методике использования метода проектов в программировании.

Этап захватывает 3-5 курсы педагогического вуза. Он базируется на курсах методики обучения информатике и спецкурсах.

Обучение методике идет по следующему содержанию:

1. Сущность метода проектов.

2. Понятие программного проекта. Типология проектов.
3. Разработка спецификаций проектов.
4. Принципы создания программных проектов.
5. Технология применения метода проектов в обучении программированию.
6. Примеры проектов в школьной практике.
7. Проект в системе уроков.
8. Проекты во внеурочной деятельности.

Обучение планированию процесса обучения информатике на основе метода проектов предполагает составление общей структуры работы (Таблица 7).

Следует особое внимание уделить разработке сценариев защиты проекта группы.

Возможны следующие сценарии (список их далеко не полный):

- Защита группы по традиционному сценарию.

Традиционный сценарий предполагает следующую схему: выступление докладчика (формулировка проблемы, гипотезы, основных характеристик программного проекта и т.д.) → демонстрация проекта → выступление оппонентов → вопросы других групп.

- Защита группы в виде конкурса.

Такой сценарий предназначен для творческих проектов и осуществляется путем организации конкурса программных проектов. Подведение итогов происходит на заседании судейской коллегии, выбранной из числа старшеклассников или других учителей. Необходимость присутствия на самом заседании членов группы, разрабатывающих данный проект, решается жюри до заседания. Каждый проект при этом может быть отмечен в разных номинациях.

Таблица 7

Структура уроков при обучении программированию на основе использования метода проектов

№	Цели урока	Используемые методы, приемы	Результат урока
1.	<ul style="list-style-type: none"> • Постановка проблемы • Разбор спецификации • Формулировка гипотез • Создание группы программистов. Определение их обязанностей 	<ul style="list-style-type: none"> • Проблемный метод • Беседа с использованием ранее разработанного учителем или старшими ребятами проекта • Выявление общих вопросов, подлежащих исследованию • Выделение связей между отдельными компонентами спецификации проекта 	<ul style="list-style-type: none"> • Проблема (далее следует формулировка проблемы) • Устный или письменный анализ спецификации проекта (далее должны быть выделены основные положения спецификации) • Особенности проекта • Гипотезы создания проекта (далее описать все особенности)
2.	<ul style="list-style-type: none"> • Определение направлений работы группы по решению поставленных задач • Разбиение проекта на отдельные блоки 	<ul style="list-style-type: none"> • Дискуссия по направлениям работ • Мозговой штурм по выявлению отдельных компонентов проекта 	<ul style="list-style-type: none"> • Направления работы (далее перечисляются основные направления) • Список блоков задач
3.	<ul style="list-style-type: none"> • Определение источников информации • Непосредственная работа над проектом 	<ul style="list-style-type: none"> • Предложение учителем списка литературы, источников INTERNET, назначение основных консультантов из старшекласников • Самостоятельная работа над проектом 	<ul style="list-style-type: none"> • Список литературы (далее пишется список в зависимости от проекта) • Рабочие версии программного проекта
4.	<ul style="list-style-type: none"> • Защита проекта 	<ul style="list-style-type: none"> • Выступление с докладом с иллюстрацией основных возможностей проекта • Дискуссия 	<ul style="list-style-type: none"> • Проект • Доклады выступающих • Сравнительная характеристика полученных результатов

- Защита проекта в виде деловой игры.

Учащимся разных групп предлагается без предварительного прослушивания доклада по соответственным проектам приступить к их изучению. При этом обращать основное внимание на ошибках программного характера. От каждой группы к своему проекту может быть выделен консультант.

Список сценариев защиты проекта можно продолжить, но лучше предложить будущим учителям информатики сделать это самостоятельно.

На практических и лабораторных занятиях по МПИ при изучении методики использования метода проектов в обучении программированию в школьном курсе необходимо не только рассмотреть с дидактической стороны все созданные до этого студентом модели, но и составить спецификации, примерное планирование в зависимости от возраста и подготовленности учащихся спецификаций проектов.

Большое влияние на формирование навыков обучения проектной деятельности учащихся оказывает педагогическая практика. В качестве одного из заданий по информатике можно для студентов предложить написание со школьниками небольшого проекта, который может быть представлен в дальнейшем на конкурс школьных работ, проводимых в рамках сотрудничества "Школа-ВУЗ".

Выводы по главе 2:

1. По результатам анализа целеполагания в процессе обучения, основанном на методе проектов, сделаны выводы о перераспределении значимости и повышения иерархической взаимосвязанности мотивационных блоков. Особую важность при этом приобретает построение первичного блока. В последующих, вложенных блоках, учитель может частично опираться на мотивы, заложенные в первом и во всех предыдущих блоках. В

результате данного процесса значительно повышается эффективность деятельности учителя, направленная на выделение целей учебного познания.

2. Опираясь на особенности формирования познавательных потребностей, была составлена модель освоения обучаемыми произвольного информационного блока. По результатам ее анализа становится возможным, исходя из выведенных соотношений, разбить весь теоретический материал на блоки, соответствующие этапам построения проекта.

3. Основными структурными элементами спецификаций проектов являются: проблема, актуальность, цели, задачи, тип, структура, методы создания программного комплекса (если это необходимо для данного типа проекта), время работы над проектом, формы представления результатов.

4. Сделаны выводы о целесообразности выделения трех временных стадий разработки спецификации проекта, характеризующих смещения приоритетов главенствования отдельных субъектов обучения в данном процессе.

5. Выделение уровней готовности школьников к осуществлению проектной деятельности (начальный, средний и продвинутый) и определение соответствующих им критериев позволяет учителю дифференцировать подход к обучению старшеклассников программированию при изучении информатики.

6. Обучение студентов педагогических вузов методике организации учебного процесса на основе использования метода проектов состоит из двух этапов, первый из которых охватывает 1-2 курсы и дисциплины предметного блока, связанные с программированием (ПРЗ на ЭВМ, программирование, ЧМ и т.д.) и второй – 3-5 курсы (методика обучения информатике).

7. Разработанные требования к подготовке учителей информатики для обучения школьников старших классов программированию на основе метода проектов позволяет определить содержание этой подготовки (сущ-

ность метода проектов, принципы создания программных проектов, понятие программного проекта, типология проектов, разработка спецификаций проектов; методика применения метода проектов в обучении старшеклассников программированию и реализация на ее основе педагогической технологии и др.).

ГЛАВА 3. МЕТОДИКА ПРОВЕДЕНИЯ И РЕЗУЛЬТАТЫ ПЕДАГОГИЧЕСКОГО ЭКСПЕРИМЕНТА

В третьей главе представлены результаты проведения педагогического эксперимента, определяющие влияние методики обучения, использованной при изучении данной дисциплины на уровень знаний, умений и навыков обучаемых по программированию, а также на уровень развития их познавательных потребностей. Рассматриваются этапы педагогического эксперимента, выделяются цели, задачи и применяемые методы исследования для каждого этапа. Определяется содержание педагогического эксперимента, анализируются результаты его проведения.

3.1 Задачи и этапы педагогического эксперимента

Разработка подходов к проведению и планированию процесса обучения, основанного на использовании метода проектов, была осуществлена в ходе специально организованного эксперимента. Основная цель эксперимента заключалась в практической проверке научной гипотезы исследования и оценке применяемой методики. Для достижения поставленной цели были определены следующие задачи педагогического эксперимента:

- 1) выявить влияние разработанной методики на развитие познавательных потребностей обучаемого;
- 2) определить эффективность предложенной методики для повышения уровня знаний, умений и навыков обучаемых в области программирования;

Экспериментальная проверка эффективности предложенной методики осуществлялась сравнением данных, полученных при обучении учащихся контрольных и экспериментальных групп путем:

- проверки качества знаний при тестировании обучаемых;
- сопоставления динамики развития умений и навыков программирования в контрольных и экспериментальных классах (группах);
- оценки динамики развития мотивационной сферы учащихся по результатам педагогического консилиума.

При организации педагогического эксперимента мы столкнулись с рядом трудностей. Первое затруднение заключалось в том, что на результаты педагогического эксперимента оказывают влияние различные факторы: особенности личности учителя, взаимоотношения учителя и обучаемых, отличия в общем умственном развитии школьников (студентов), нравственная атмосфера в классе (группе) и т.д. Для преодоления этой трудности контрольные и экспериментальные группы подбирались таким образом, чтобы все внешние воздействия были примерно одинаковы, а отличия заключались в тех факторах, действия которых нами исследуется. Кроме этого, применялся метод «перекрещивающихся» групп, при котором исследуемый нами фактор действовал сначала в одной группе, потом в другой.

С этой же целью нами применялся метод единственного отличия. Приведем его схематическое представление:

$$A, B, V, \Gamma \rightarrow D, E$$

$$A, B, V, (\Gamma+) \rightarrow D, (E+),$$

где Γ – неизменяемый параметр, а $(\Gamma+)$ – изменяемый параметр исследуемого нами фактора по предлагаемой методике. При изменении параметра Γ (при неизменных остальных параметрах), мы наблюдали изменение параметра E , что позволяет говорить о Γ как о причине изменения E [53, с. 28]. При прочих равных условиях мы наблюдали статистически достоверную положительную динамику изменения глубины усвоения знаний учеников экспериментальной группы, чем в контрольной. Это позволило нам сделать вывод о их взаимосвязи.

Следующая трудность заключается в том, что с одной стороны для статистической достоверности требуется достаточно большое число испытуемых, а с другой стороны, большая массовость не позволяет достаточно глубоко и полно проследить трудно наблюдаемые изменения, связанные с мыслительной деятельностью. Для преодоления этого затруднения мы использовали метод репрезентативного отбора и воспользовались таблицей больших чисел. Для получения статистически достоверных результатов проведения эксперимента, количество испытуемых () взято согласно принятому уровню значимости $E/P=0,95$, где E – допустимая ошибка, P – минимальное количества испытаний для различных уровней погрешности [52].

Для более глубокого исследования и изучения отличий в результатах, контрольных и экспериментальных групп мы применяли методы вторичной статистической обработки. Чтобы быть уверенными в том, что все изменения в отобранной нами сравнительно небольшой группе из общего числа будут объективно совпадать с изменениями, происходящими со всеми участниками эксперимента, отбор обучаемых для исследования осуществлялся следующим образом: общие распределения изучаемых признаков контрольной и экспериментальной группы до проведения эксперимента были одинаковыми.

Объектами наблюдения в эксперименте являлись учащиеся старших классов школ г. Шадринска и Шадринского района, а также студенты факультетов информатики и физико-математического Шадринского государственного педагогического института (ШГПИ).

Исследование проводилось в три этапа.

На этапе констатирующего эксперимента выявлялись предпосылки реализации метода проектов в процессе обучения программированию. Этому этапу дидактического эксперимента по времени соответствовало формирование гипотезы и задач исследования. Основная цель первого эта-

па – получить материал для дальнейшего теоретического осмысления. Основными методами первого этапа были: наблюдение за деятельностью учителей и обучаемых, опросы, анкетирование, срезы знаний, тестирование (Таблица 8).

Второй этап, **поисковый эксперимент**, характеризовался внедрением и исследованием метода проектов в процесс обучения программированию. Данный этап сопровождался поиском методических средств и приемов реализации этого процесса (Таблица 9).

Третий этап, **формирующий эксперимент**, был направлен на сопоставление прогнозируемых результатов с результатами практического внедрения, на разработку критериев эффективности предложенной методики обучения и, соответственно, оценку результатов и внесение корректив в исходную рабочую гипотезу (Таблица 10).

Оценка эффективности предложенной методики осуществлялась с применением математической статистики.

Для проверки гипотез проводился расчет по коэффициенту Пирсона χ^2 . Вычислялась величина $T_{\text{наб}}$, характеризующая степень различия по данному показателю в рамках полугодия при традиционном обучении. Критическое значение статистики ($T_{\text{кр}}$) – допустимый интервал различия между показателями, при котором обе рассматриваемые выборки объявляются принадлежащими одному классу. Гипотеза H_0 принимается, если $T_{\text{наб}} \leq T_{\text{кр}}$, если же $T_{\text{наб}} > T_{\text{кр}}$ принимается гипотеза H_1 . Критическое значение статистики, используемое для уровня значимости 0,05 и числа степеней свободы – $4-1=3$ по таблице составляет 7,815.

Первый этап. Констатирующий эксперимент

Цели и задачи этапа	Методы и способы педагогического исследования	Методы, повышающие достоверность	Результаты исследования
<ul style="list-style-type: none"> Выявление основных проблем обучения программистов в школе и педагогическом вузе Определение уровня знаний, умений и навыков по программированию школьников старших классов и студентов педагогических вузов Разработка способов организации процесса обучения программистов, направленного на создание динамичной системы знаний по программированию Изучение опыта работы учителей и преподавателей в направлении решения проблемы мотивации учения школьников и студентов 	<ul style="list-style-type: none"> Теоретический анализ литературы по проблеме исследования Индивидуальная экспертная оценка Групповая экспертиза Изучение передового опыта работы учителей. Анализ конкретных экспериментальных ситуаций Наблюдение. Беседа. Анкетирование студентов и преподавателей. Тестирование 	<ul style="list-style-type: none"> Обоснование выбора экспертов. Обеспечение репрезентативности выборки групп студентов Определение полноты изучения литературы (сравнение с полной библиографией по изучаемой проблеме) Проверка одной и той же методики в разных классах Апробация результатов исследования в форме выступлений перед преподавателями, аспирантами, учителями Использование комплекса средств для получения достоверности результатов (повторяемость эксперимента, его продолжительность) 	<ul style="list-style-type: none"> Выявлены перспективы использования метода проектов в формировании мотивационной сферы обучаемых Выделены виды проектов и определены основные положения методики использования метода проектов Разработаны первоначальные методические рекомендации по использованию метода проектов по языкам программирования (на основе языка Pascal) Определен критерий эффективности использования метода проектов в зависимости от возрастных особенностей обучаемых, темы и т.д.

Таблица 9

Второй этап. Поискный эксперимент

Цели и задачи этапа	Методы и способы педагогического исследования	Методы, повышающие достоверность	Результаты исследования
<ul style="list-style-type: none"> • Выбор групп обучаемых с примерно одинаковым уровнем владения знаниями, умениями и навыками по программированию • Определение объективной возможности введения метода проектов в методику обучения программированию • Разработка методики, направленной на развитие мотивационной сферы обучаемых • Разработка принципов построения спецификаций проектов, методов их создания, типологии, перспектив и возможностей 	<ul style="list-style-type: none"> • Теоретический анализ возможностей языка программирования, анализ структурных единиц языка, конкретных методов и приемов программирования • Изучение и обобщение опыта работы учителей и преподавателей 	<ul style="list-style-type: none"> • Обоснование выбора экспериментальных групп • Согласование инвариантного ядра содержания метода проектов с вариативной оболочкой • Определение уровня заинтересованности обучаемых на основе представления им выбора содержания учебного материала 	<ul style="list-style-type: none"> • Сделано предположение, что одним из возможных способов решения проблемы развития мотивационной сферы обучаемых в процессе обучения программированию может быть введение в него метода проектов • Разработаны основные требования к проектам, их свойствам, типологии, спецификации • Выделены методические рекомендации по их использованию при изучении программирования

Таблица 10

Третий этап. Формирующий эксперимент

Цели и задачи этапа	Методы и способы педагогического исследования	Методы, повышающие достоверность	Результаты исследования
<ul style="list-style-type: none"> • Уточнение методики применения метода проектов при изучении языков и методов программирования • Определение влияния разработанной методики на развитие познавательных потребностей личности • Практическое внедрение предложенной методики 	<ul style="list-style-type: none"> • Метод статистической обработки результатов педагогического эксперимента • Изучение и анализ результатов деятельности учеников и учителей • Анкетирование, интервью и беседы с обучающимися, учителями, преподавателями • Сравнительный анализ. 	<ul style="list-style-type: none"> • Применение метода единственного отличия • Применение метода переключивающихся групп • Длительность наблюдения (1998 – 2000 г.г.) 	<ul style="list-style-type: none"> • Установлена связь предложенной методики обучения программированию с дидактикой овладения знаниями, умениями и навыками программированию • Выявлены формы и методы построения процесса организации обучения по методу проектов, способствующие развитию познавательных потребностей личности

Критерий Пирсона $T_{\text{наб}}$ вычисляется по формуле:

$$T_{\text{наб}} = \frac{1}{n_1 \cdot n_2} \sum_{i=1}^c \frac{(n_1 Q_{2i} - n_2 Q_{1i})^2}{Q_{1i} + Q_{2i}},$$

где n_1 – число членов контрольной группы, n_2 – число членов экспериментальной группы, Q_{1i} – число членов экспериментальной группы, достигших i -го уровня, Q_{2i} – число членов контрольной группы, достигших i -го уровня, c – число выделенных уровней.

3.2 Содержание и результаты педагогического эксперимента

Констатирующий педагогический эксперимент (1995-1996гг) был посвящен выявлению предпосылок реализации метода проектов в процессе обучения программированию. Этому этапу педагогического эксперимента по времени соответствовало формирование гипотезы и задач исследования, поиск путей их решения. Основная цель первого этапа – получить материал для дальнейшего теоретического осмысления.

На данном этапе проводились пробные занятия со школьниками старших классов и студентами физико-математического факультета и факультета информатики с целью проверки возможности повышения качества знаний и развития познавательных потребностей обучаемых. Учителям и преподавателям информатики была предложена анкета, представленная в приложении 1.

Вопросы анкеты посвящены анализу проблемы выбора методов обучения программированию.

Из 18 учителей школ города и района, 17 преподавателей ШГПИ, ответивших на вопросы анкеты, лишь 15 считают себя компетентными и достигшими высокого уровня в программировании. Они отмечают, что

уровень их владения методами программирования низок из-за быстрых смен приоритетов в стратегии программирования (смена операционных систем, высокий динамизм в развитии языков программирования, переход от процедурного к объектно-ориентированному программированию, разработка систем RAD и т.д.).

Определение уровня знаний по программированию у учителей и преподавателей было организовано в виде схемы-теста, представленного в приложении 2.

Анализируя результаты выполнения схемы-теста, можно сделать вывод о недостаточной компетентности учителей информатики в программировании. Необходимо отметить, что наиболее слабые результаты были у тех учителей, которые не имеют возможности вести преподавание на новейшей современной технике (в их распоряжении лишь компьютеры "Корвет", "Агат", "Электроника", "Ямаха" и IBM 86-286).

Анкетирование среди учащихся и студентов (анкета в приложении 3) выявило следующие результаты: среди 176 учеников и 141 студента, принявших участие в анкетировании, 127 указали на необходимость использования метода проекта в старших классах общеобразовательной школы и в вузе. Среди 317 опрошенных 97 обучались уже на некоторых этапах по методу проектов. Они отмечают, что им нравится самостоятельно создавать большие программные комплексы. "Работа над проектом позволяет увидеть перспективные возможности языка программирования, убедиться в необходимости изучения отдельных методов программирования" (Тигиняну Е.). Отдельные учащиеся уже имеют за плечами большой опыт по созданию программных комплексов, например, Попов Е. (381 гр., факультет информатики ШГПИ) в школе: "Тетрис" (basic), в институте: графический редактор (Pascal), текстовый редактор (Pascal), игра "Балда" (Pascal), графический редактор (Delphi), школьный алгоритмический язык (Delphi), программа, помогающая создавать тесты различного уровня сложности по

разным предметам (Delphi). Реализация такого количества проектов на высоком научном уровне позволило ему стать президентом Студенческой Академии ШГПИ, принять участие в региональных олимпиадах и фестивале-конкурсе научного и прикладного творчества молодежи в г. Кургане. Среди самых интересных проектов студенты выделили а) творческий проект "Интернет-газета" по изучению языка HTML (45 %); б) графический редактор (язык программирования Delphi) (16 %); в) база данных по студентам (язык программирования Pascal) (15 %); г) групповой творческий проект, например, "Карта больного", "Магазин", "Гороскоп" (14 %).

Многие из школьников и студентов отмечали, что создание проектов позволяет «разные части теории языков и методов программирования соединять вместе. Например, работа с файлами и строковые переменные, создание интерфейса, графика, текстовый режим работы и т.д.» (Журило Л.).

Проведенное тестирование в начале эксперимента показало отсутствие четкой системы по программированию у обучаемых, изучающих информатику. Они имеют расчлененные представления о структуре этой системы в виде непересекающихся множеств дискретного характера. При создании некоторого взаимодействия нескольких множеств при решении задач появляется новое множество, включающее в себя это взаимодействие, но отличное от тех, из которых оно было создано.

После проведенных анкетирования, контрольных срезов, тестов, посещения занятий с традиционной методикой проведения, так и с использованием метода проектов были сделаны следующие выводы:

- для эффективного обучения школьников программированию необходимо разработать методику обучения информатике на основе метода проектов;
- для успешной реализации метода проектов требуется разработка педагогической технологии, технологическая схема и технологическая

карта которой учитывает специфику применения метода проектов в обучении программированию в курсе информатики;

- использование метода проектов в процессе обучения программированию является эффективным средством развития познавательных потребностей учащихся.

Чтобы обеспечить решение этих проблем, мы остановились на разработке методики обучения программированию, основанной на методе проектов. Результатом первого этапа исследования стало становление теоретической базы исследования и разработка гипотезы.

Поисковый педагогический эксперимент (1996-1998гг) был посвящен проверке основных положений, по которым осуществлялся педагогический эксперимент, корректировалась предлагаемая методика.

На втором этапе исследования была разработана программа курса программирования на основе изучения языков программирования Basic, Pascal и Object Pascal в среде Delphi, разработана методика проведения занятий, построенных на основе использования метода проектов. При ее создании мы опирались на уже существующие программы и собственный опыт работы.

Разработанная методика обучения программированию апробировалась в школах № 1 и 4 в специализированных классах и классах, работающих по традиционной программе, в рамках курсов "Языки и методы программирования» (ЯиМП) и методики обучения информатике на факультете информатики и физико-математическом ШГПИ. Для проверки нашей гипотезы мы выбрали следующие классы (группы) – 11в (специализация – филология, школа № 1), 10б (специализация – английский язык и информатика, школа №4), 281 и 283 группы (специализация – английский язык и информатика, ШГПИ).

Для определения уровня знаний, умений и навыков обучаемым предлагались контрольные задания, для которых определялся операцион-

ный и процессуальный состав выполняемых действий. Анализировались результаты 4-х контрольных срезов, проведенных по окончании изучения соответствующей темы.

Комплексный анализ динамики изменения уровня знаний, умений и навыков мы проводили на основе оценки решения заданий трех видов. Первый блок заданий включал в себя вопросы теоретического характера, направленные на проверку усвоения знаний (ФЗ - фонд знаний). Второй блок содержал набор практических задач, ориентированных на выявление уровня применимости теоретических знаний в решении проблем прикладного характера (ПЗ – проверочные задания). Третий блок был предназначен для проверки умений решать задачи, опирающиеся на выявление внутрипредметных связей (ЗСБ – задания на связь между блоками знаний).

При оценке ответов учащихся, мы пользовались пятибалльной шкалой. Учитывая мониторинговую зависимость результата от сложности заданий, каждому блоку был присвоен коэффициент сложности.

Уровень знаний, умений и навыков обучаемых определялся с помощью числового значения M по формуле:

$$M = \frac{O_1 \cdot k_1 + O_2 \cdot k_2 + O_3 \cdot k_3}{5 \cdot (k_1 + k_2 + k_3)} \cdot 100\%,$$

где O_i - оценка результатов деятельности обучаемых по выполнению заданий i -го блока, k_i - коэффициент значимости этого блока. Для удобства обработки данных мы установили сумму коэффициентов значимости на каждом этапе в значение 10. Сами коэффициенты значимости определялись с помощью экспертной оценки.

Значения M в диапазоне 95-100% соответствовали высшему (отличному) уровню знаний, умений и навыков, 80-95% - хорошему уровню, 65%-80%, - удовлетворительному уровню. При $M < 65\%$ учащиеся соответствовали неудовлетворительному уровню.

Результаты первого и последнего контрольных срезов представлены ниже (Таблица 11).

Полученные результаты наглядно показаны на диаграмме (Рис. 13).

Анализ результатов контрольных срезов показал, что при использовании экспериментальной методики наблюдается положительный рост уровня знаний, умений и навыков обучаемых. Это позволило сделать вывод о преимуществе внедрения метода проектов в процесс обучения программированию по сравнению с традиционными методами.

Формирующий педагогический эксперимент (1998-2000 г.г.) был проведен в школах № 1 и № 4 г. Шадринска и на факультетах информатики и физико-математическом ШГПИ. В процессе проведения формирующего эксперимента реализовывалась методика обучения программированию, направленная на использование метода проектов.

Были выделены экспериментальные и контрольные группы с составом обучаемых примерно одинаковым по успеваемости и способностям.

Мы выбрали школьников 10-11 классов и студентов второго-третьего курсов, распределив их следующим образом:

- школа № 1, язык программирования Pascal, три класса: 11а (специализация – физика-информатика), 11б (специализация – экономика – информатика), 11в (специализация – филология) первые подгруппы обучались по методу проектов, вторые же подгруппы этих же классов – по традиционной методике;

- школа № 4, язык программирования Pascal, три класса: 11а (специализация – информатика и экономика), 10б (специализация – английский язык и информатика), 10в (без специализации) первые подгруппы обучались по методу проектов, вторые же подгруппы этих же классов – по традиционной методике;

Таблица 11

Результаты контрольных срезов в поисковом эксперименте

Условные обозначения:

ФЗ – фонд знаний (результаты ответа на вопросы теоретического блока);

ПЗ – проверочные задания (результаты решения задач, направленных на проверку текущих знаний);

ЗСБ – задания на связь между блоками знаний (результаты решения задач, направленных на установление взаимосвязи между отдельными блоками знаний).

Группа	Средний балл за ответы на вопросы						Средний балл за работу	Распределение учащихся по уровням			к-во учащихся, достигших отличного уровня (%)	к-во учащихся, достигших хорошего уровня и выше (%)	к-во учащихся, достигших удовлетворительного уровня и выше (%)	
	ФЗ		ПЗ		ЗСБ			Отл	Хор	Удовл				
	балл	коэф знач	Балл	коэф знач	балл	Коеф знач								
Результаты первого контрольного среза														
11в (№1)	3,7	6	3,3	2	3,1	2	3,37	3	7	11	3	12,50%	41,67%	87,50%
10а (№4)	3,6	4	3,4	2	3,3	4	3,43	5	9	9	2	20,00%	56,00%	92,00%
281, 283	3,5		3,7		3,5		3,57	4	10	6	2	18,18%	63,64%	90,91%
Результаты четвертого контрольного среза														
11в (№1)	3,7	5	3,7	2	3,5	3	3,63	5	9	8	2	20,83%	58,33%	91,67%
10а (№4)	3,9	4	4,3	3	3,8	3	4,00	6	13	5	1	24,00%	76,00%	96,00%
281, 283	4,1		4,0		3,9		4,00	7	9	4	2	31,82%	72,73%	90,91%

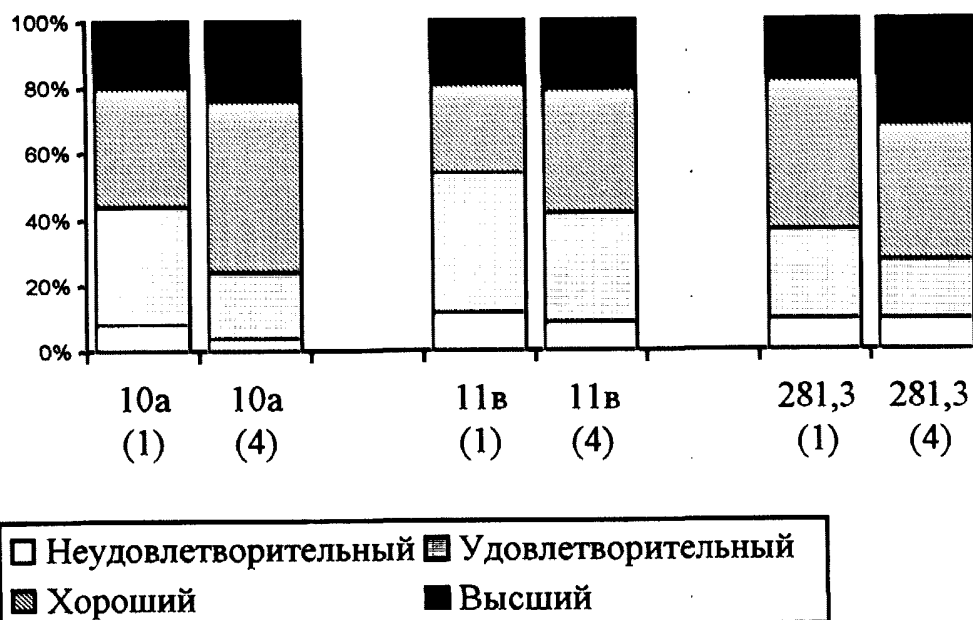


Рис. 13. Процентное соотношение уровней знаний, умений и навыков по результатам поискового эксперимента

- физико-математический факультет, третий курс, язык программирования Pascal, три подгруппы (301 «а», 302 «а», 303 «а») обучались по методу проектов и три (301 «б», 302 «б», 303 «б») – по традиционной методике;

- факультет информатики, второй курс, язык программирования Pascal, три группы (281, 283, 286) обучались по методу проектов, три группы (282, 284, 285) – по традиционной методике.

Среди 621 испытуемого суммарный объем экспериментальной группы (с учетом обновления состава по окончании очередного учебного года) составил 327, контрольной – 294 обучаемых.

Таблица 12

Результаты контрольных срезов

Условные обозначения:

ФЗ – фонд знаний (результаты ответа на вопросы теоретического блока);

ПЗ – проверочные задания (результаты решения задач, направленных на проверку текущих знаний);

ЗСБ – задания на связь между блоками знаний (результаты решения задач, направленных на установление взаимосвязи между отдельными блоками знаний).

"НУ" – неудовлетворительный уровень, "У" – удовлетворительный, "Х" – хороший, "О" – отличный (отличный).

Группа	Средний балл за ответы на вопросы						Средний балл за работу	Распределение учащихся по уровням				к-во учащихся, достигших отличного уровня (%)	к-во учащихся, достигших хороше-го уровня и выше (%)	к-во учащихся, достигших уровня и выше (%)
	ФЗ		ПЗ		ЗСБ			О	Хор.	Удовл.	Неуд.			
	балл	коэф знач	балл	коэф знач	балл	коэф знач								
Контр.	3,2	5	3,6	3	2,9	2	3,23	38	71	168	17	12,93%	37,07%	94,22%
Эксп.	3,3		3,6		2,7		3,20	43	86	179	19	13,15%	39,45%	94,19%
Результаты первого контрольного среза														
Контр.	3,4	4	3,3	4	3,1	2	3,27	41	62	176	15	13,95%	35,03%	94,90%
Эксп.	3,9		4,3		3,3		3,83	52	91	165	19	15,90%	43,73%	94,19%
Результаты шестого контрольного среза														

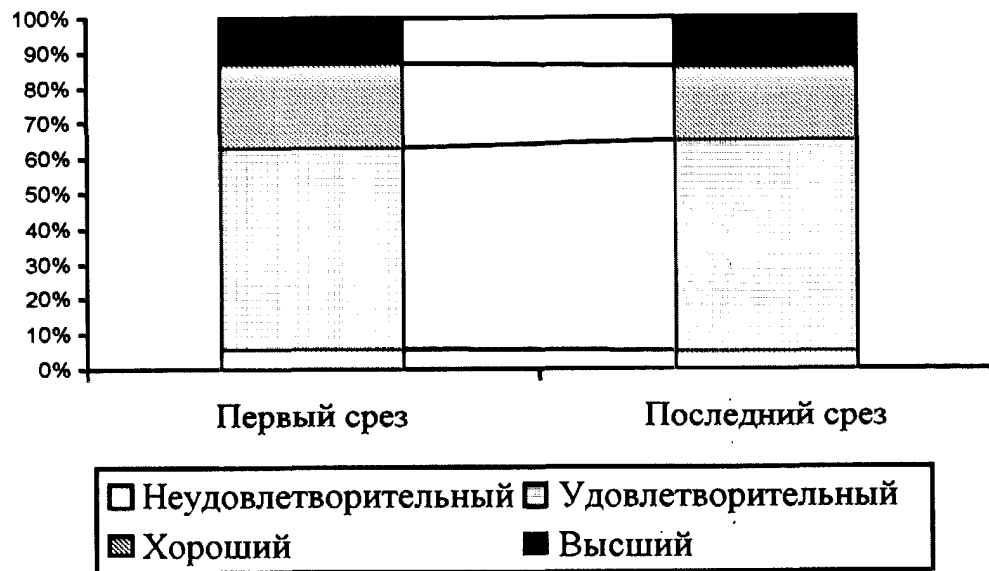


Рис. 14. Соотношение уровней знаний, умений и навыков обучаемых контрольной группы

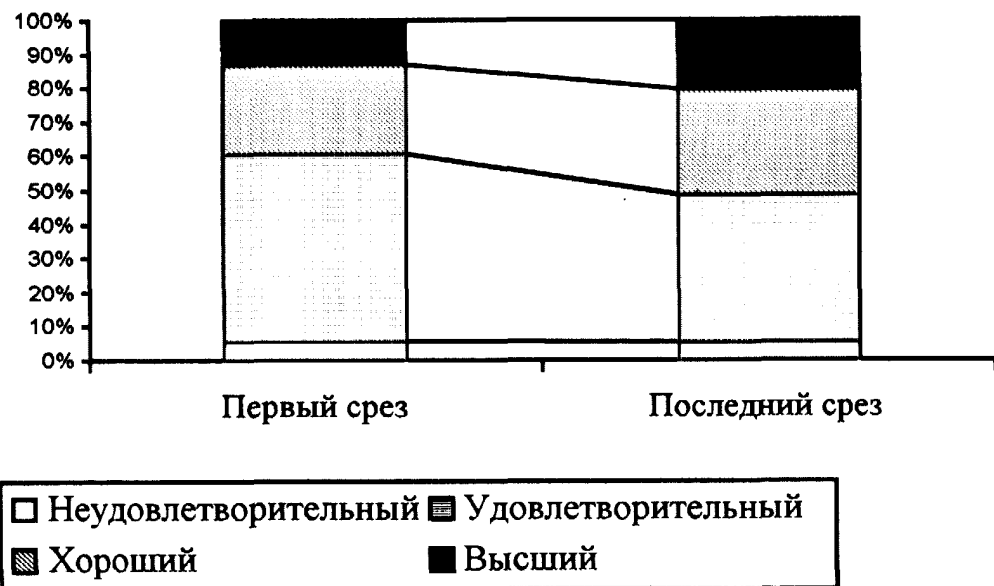


Рис. 15. Соотношение уровней знаний, умений и навыков обучаемых экспериментальной группы

В экспериментальных группах учителя использовали программу курса "Информатика", раздел "Алгоритмизация и программирование" на основе разработанной нами методики. В педагогическом вузе мы применяли данную методику в курсе "Языки и методы программирования".

Нами было проведено 6 контрольных срезов, позволивших проследить динамику изменения в уровнях знаний, умений и навыков

Полученные результаты представлены в таблице и диаграммах выше (Таблица 12, Рис. 14, Рис. 15).

Анализ полученных результатов показывает наличие положительной динамики роста уровня знаний, умений и навыков по программированию при использовании метода проектов в образовательном процессе.

Систематическая работа на занятиях по предложенной методике в экспериментальных группах показала, что изменение числа обучаемых, достигших высокого уровня владения знаниями, умениями и навыками по программированию, происходит быстрее, чем в контрольном классе. Результаты проведенного эксперимента свидетельствуют о положительном влиянии предложенной методики на уровень знаний, умений и навыков студентов и учащихся.

Таблица 13

Результаты распределения обучаемых по уровням владения знаниями, умениями и навыками по программированию

Тип обучения	Всего обучаемых	Количество учащихся, достигших соответственного уровня				Т _{наб}
		Высший	Хороший	Удовлетворительный	Неудовлетворительный	
Экспер.	327	52	91	165	19	12,92
Контр.	294	41	62	176	15	

Из факта $T_{\text{набл}} > T_{\text{крит}}$ ($12,92 > 7,85$) мы сделали вывод, что с достоверностью 95% достижение уровней сформированности знаний по программированию учащихся контрольных и экспериментальных классов обусловлены не случайными факторами, а носят закономерный характер. Причиной этого явилось использование метода проектов в учебном процессе экспериментальных классов.

Следующим этапом в проверке эффективности предложенной методики было определение изменений в мотивационной сфере учащихся. Оценка эффективности влияния технологии использования метода проектов в обучении программированию на развитие познавательных потребностей проводилось по следующим показателям (Таблица 14):

- осознание необходимости включения в учебно-познавательную деятельность;
- готовность к учебно-познавательной деятельности.
- проявление инициативы;
- удовлетворенность процессом учебного труда и его результатами;

Исходя из этих критериев нами была составлена диагностическая карта (Рис. 16) для определения уровня развития познавательных потребностей учащихся в процессе обучения программированию.

В нашем исследовании оценка выраженности познавательной потребности производилась в системе набранного количества баллов по шкале:

- 0 – полностью отсутствует тот или иной элемент, характеризующий качество в проявлении познавательной потребности;
- 1 – элемент частично присутствует;
- 2 – элемент присутствует полностью.

Диагностическая карта заполнялась экспертным советом на каждом контрольном срезе в течении как поискового, так и формирующего экспериментов.

Основные параметры развития познавательных потребностей у учащихся

Показатели	Критерии
Осознание необходимости включения в учебно-познавательную деятельность	1. Наличие личной познавательной цели – познавательная цель преследует удовлетворение личной потребности узнавать что-то новое, изучить определенный предмет или явление 2. Наличие коллективно-значимой познавательной цели 3. Интерес к решению учебной задачи, проблемы или противоречия 4. Желание достичь цели, может быть и трудной 5. Влечение к поиску оригинальных способов разрешения возникших противоречий
Готовность к учебно-познавательной деятельности	6. Самостоятельный перенос ранее усвоенных знаний и умений в новую ситуацию 7. Умение самостоятельно найти проблему в привычной ситуации 8. Видение альтернатив, вариантов решения 9. Комбинирование ранее известных способов в новый 10. Владение приемами самостоятельного познания
Проявление инициативы	11. Инициативная исполнительность (созданы условия, требующие проявления инициативы в процессе учебного труда) 12. Самостоятельность в выдвижении гипотез, идей, способов решения 13. Готовность к разрешению противоречий, учебных задач, проблем
Удовлетворенность процессом учебного труда и его результатами	14. Увлеченность процессом решения учебных задач. 15. Переживание радости от процесса учебной деятельности 16. Испытание чувства радости от результата учебного труда

Диагностическая карта по выявлению уровня развития познавательных потребностей учащихся при включении их в учебный процесс

Ф.И.О. обучаемого _____

Класс (группа) _____

Школа (факультет) _____

Дата обследования	Осознание необходимости включения в учебно-познавательную деятельность					Готовность к учебно-познавательной деятельности					Проявление инициативы			Удовлетворенность процессом учебного труда и его результатами		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Рис. 16. Диагностическая карта по выявлению уровня развития познавательных потребностей учащихся

В состав экспертного совета входили: руководитель экспертного совета (научный руководитель), учитель, классный руководитель, ученики (по выбору учителя), родители (по выбору классного руководителя).

Задача экспертного совета состояла в определении объективно-значимой оценки динамики изменения мотивационной сферы учащихся и, в частности, познавательных потребностей.

Результаты анализа диагностических карт, собранных на протяжении формирующего эксперимента показаны ниже (Рис. 17, Рис. 18, Таблица 15)

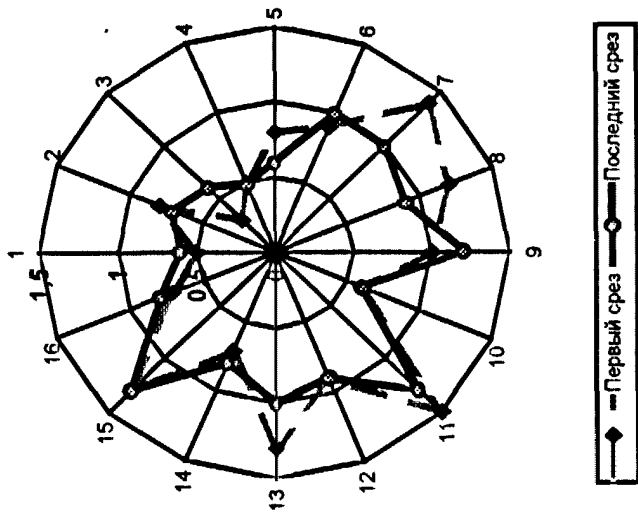


Рис. 17. Развитие познавательных потребностей обучаемых контрольной группы

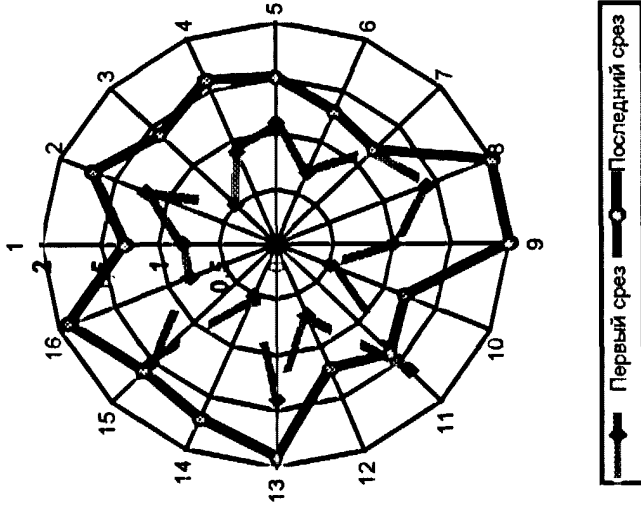


Рис. 18. Развитие познавательных потребностей обучаемых экспериментальной группы

Развитие познавательных потребностей учащихся

Тип обучения	Всего обучаемых	Уровень развития познавательных потребностей	
		Первый срез	Последний срез
Экспер.	327	52,3%	72,5%
Контр.	294	51,6%	54,1%

По результатам анализа полученных данных был сделан вывод о преимуществе разработанной методики по сравнению с традиционными методами обучения в области развития мотивационной сферы учащихся.

Выводы по главе 3.

1. Целенаправленное и систематическое применение методики обучения программированию в курсе информатики, основанной на использовании метода проектов, положительно влияет на качество приобретаемых умений и навыков у обучаемых.
2. Проведенное исследование динамики развития познавательных потребностей обучаемых позволило сделать вывод об эффективности предлагаемой методики.

ЗАКЛЮЧЕНИЕ

Результаты проведенного исследования позволяют сформулировать следующие выводы:

1. Предлагаемая методика обучения программированию в курсе информатики, основанная на применении метода проектов позволяет, разработать педагогическую технологию, обеспечивающую возрастание познавательных потребностей обучаемых и повышение эффективности процесса обучения.

2. Поскольку структура педагогической технологии базируется на использовании технологической карты и технологической схемы процесса обучения, то для разработки технологии обучения, соответствующей методике применения метода проектов, необходимо наполнение блоков технологической схемы и этапов технологической карты содержанием, отражающим специфику метода проектов в обучении программированию.

3. К типологическим признакам учебных проектов следует отнести: цели создания проекта, методы, доминирующие в процессе разработки проекта, характер творческой деятельности при разработке проекта, характер координации проекта, характер контактов и количество участников проекта, продолжительность разработки проекта.

4. Основными структурными элементами спецификаций проектов являются: проблема, актуальность, цели проекта, задачи проекта, тип проекта, структура проекта, методы создания программного комплекса (если это необходимо для данного типа проекта), время работы над проектом, формы представления результатов.

5. Выделение уровней готовности школьников к осуществлению проектной деятельности (начальный, средний и продвинутый) и определение соответствующих им критериев позволяет учителю дифференцировать

подход к обучению старшеклассников программированию при изучении информатики.

6. Разработанные требования к подготовке учителей информатики для обучения школьников старших классов программированию на основе метода проектов позволяет определить содержание этой подготовки (сущность метода проектов, понятие программного проекта, типология проектов, разработка спецификаций проектов, принципы создания программных проектов, методика применения метода проектов в обучении старшеклассников программированию и реализация на ее основе педагогической технологии и др.).

7. Проверка уровней владения знаниями, умениями, навыками, а также динамики развития познавательных потребностей в классах, изучающих программирование на основе метода проектов, позволила выявить положительное влияние применяемой методики на процесс обучения. Педагогический эксперимент показал действенность предлагаемой методики, эффективность созданной на ее основе технологии и целесообразность внедрения в обучение учащихся старших классов общеобразовательных школ программированию в курсе информатики.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Абульханова К.А. О субъекте психической деятельности. Методол. проблемы психологии. М.: Наука, 1973. 288 с.
2. Абушкин Х. Х. Проблемное обучение - учителю. - Саранск : Морд. кн. изд-во, 1996. 174 с.
3. Аванесов В.С. Применение статистических методов и ЭВМ в педагогических исследованиях // Введение в научное исследование по педагогике. - М.: Просвещение, 1988. С. 139-155.
4. Альтшуллер Г.С. Алгоритм изобретения. - М.: Московский рабочий, 1973. 296 с.
5. Амосов НМ. Моделирование сложных систем - Киев: Наукова думка, 1968. 88 с
6. Ананьев Б.Г. Психологическая структура личности и ее становление в процессе индивидуального развития человека // О проблемах современного человекознания. -М., 1977. С.187-274.
7. Андреева А.Д. Особенности отношения к учению подростков и старших школьников: Дисс. ... канд. психол. наук. - М., 1989. 160 с.
8. Андрианов П.Н. Развитие технического творчества в трудовом обучении учащихся общеобразовательных школ: Дис. ... докт. пед. наук, - М., 1985. 520с.
9. Арестова О.Н. Мотивация и перспективное целеполагание в мыслительной деятельности // Вестник МГУ, серия Психология, 1999. № 3. С. 16-25.
10. Асеев Б.Г. Мотивация поведения и формирования личности. - М.: Мысль, 1976. 158 с.
11. Асмолов А.Г. Личность как предмет психологического исследования. М., 1984. 104 с.

12. Бабанский Ю.К. Интенсификация процесса обучения. М.: Знание, 1987. 80 с.
13. Бабанский Ю.К. Оптимизация педагогического процесса: Обще-дидактический аспект. – М.: Педагогика, 1977. 256 с.
14. Бабанский Ю.К. Оптимизация процесса обучения: Аспект предупреждения неуспеваемости школьников – Ростов-на-Дону: Рн/ДГПИ, 1972. 347 с.
15. Бабанский Ю.К. Проблемы повышения эффективности педагогических исследований. - М.: Педагогика, 1982. 192 с.
16. Баннов И. И. Проектирование как условие развития личностных качеств старших школьников. -Дисс. ... канд. психологич. наук- М., 1998. 188с.
17. Бахманн П., Френцель М., Ханцшман К., Хельтциг Ф., Херриг Д., Кучке К.-Х., Мах В., Зандманн Х., Шмидт К., Шваар М., Зоботта Х. Программные системы: Пер. с нем. - М.: Мир, 1988. 288 с.
18. Беспалько В.П. Программированное обучение: Дидактические основы. М.: Высш. шк., 1970. 300 с.
19. Беспалько В.П. Слагаемые педагогической технологии. - М.: Педагогика, 1989. 282 с.
20. Блинов В.М. Эффективность обучения. / Методологический анализ определения этой категории в дидактике - М.: Педагогика, 1976. 192 с.
21. Бобко И.М., Коуров В.Г., Собакинских Н.И. Адаптивные педагогические программные средства. – Новосибирск: Изд-во Новосибирского гос. ун-та, 1991. 101 с.
22. Богоявленская Д.Б Психологические основы интеллектуальной активности Дисс ... докт. психол. наук. - М., 1988. 395 с.
23. Бодалев А.А. Психология личности - М.: МГУ, 1988. 194 с.
24. Бодалев А.А. Психология о личности. М.: Изд-во Моск. ун-та, 1988. 188 с.

25. Божович Л.И. Этапы формирования личности в онтогенезе // Вопросы психологии, 1978. № 4. С. 23-25.
26. Бондаревская Е.В. Гуманистическая парадигма личностно ориентированного образования // Педагогика, 1997. № 4. С. 38-39.
27. Боно Э. Рождение новой идеи. - М.: Прогресс, 1976. 143 с.
28. Боткин Дж. Инновационное обучение, микроэлектроника и интуиция. // Перспективы: вопросы образования. - Париж, 1983, № 1. - С. 39-47.
29. Брагин В.Я. Вопросы теории и методики изучения раздела "проект" в курсе "Технология". - Пермь, 1995. 56 с.
30. Брукс Ф.П. мл. Как проектируются и создаются программные комплексы: Пер. с англ. - М.: Наука, 1979. 152 с.
31. Брунер Д. Процесс обучения. - М.: изд-во АПН РСФСР, 1962. 83 с.
32. Брушлинский А. В. Психология мышления и кибернетика. - М.: Мысль, 1970. 189 с.
33. Брушлинский А.В. Мышление: процесс, деятельность, общение. - М.: Наука, 1982. 286 с.
34. Брушлинский А.В. Субъект: мышление, учение, воображение. - Москва-Воронеж: НПО «МОДЭК», 1996. 396 с.
35. Вамош Т. Приоритет человеческого фактора // Перспективы: Вопросы образования, 1988. № 3. С. 39-45.
36. Воронцов Б.Н. Разумные потребности личности: сущность, критерий, пути формирования / Воронежский гос. ун-т, Воронеж, 1990. 190с.
37. Габай Т.В., Талызина Н.Ф. Пути и возможности автоматизации учебного процесса. - М.: Знание, 1977. 196 с.
38. Гаспарский В. Праксеологический анализ проектных разработок. - М.: Мир, 1978. 172 с.
39. Гварамия Г., Маргвелашвили И., Мосиашвили Л. Опыт разработки компьютерных учебных пособий по физике // Информатика и образование, 1990. № 6. С. 79-81.

40. Гегель Г. Собрание сочинений в 13 томах. Том I. - М.: Росиздат, 1929. 368 с.
41. Геллерштейн С.Г. Вопросы психологии труда // Психологическая наука в СССР, том II - М.: Изд-во АПН РСФСР, 1960. С. 337-361
42. Геллерштейн С.Г. Психологический анализ трудовой деятельности в свете задач инженерной психологии // Проблемы инженерной психологии Выпуск I - М.: Научный совет АН СССР по проблемам конкретных социальных исследований, 1968. С. 156-161.
43. Генов Ф. Психология управления. Пер. с болгарского. М.: Прогресс, 1982. 422 с.
44. Георгиева Т.С. Высшая школа США на современном этапе – М.: Высшая школа, 1989. 80 с.
45. Гергей Г. Машбиц Е. Психолого-педагогические проблемы эффективного применения компьютеров в учебном процессе // Вопросы психологии, 1985. № 3. С. 41-49.
46. Гершензон В.Е., Зейналова Е.Ю., Кошевой В.А., Левченко И.В., Тараканова О.Н. Проект "подсолнух" - образ новой обучающей среды // Информатика и образование, 1993. № 5. С. 91-93.
47. Гершунский Б. С. Никандров Н.Д. Методологическое знание в педагогике. М.: Знание. 1986. 311 с.
48. Гершунский Б.С. Компьютеризация в сфере образования: проблемы и перспективы.– М.: Педагогика, 1987. 264 с.
49. Гончарова С.В. Использование компьютерных моделей для развития творчества учащихся // Развитие творческой активности учащихся в процессе обучения и профессиональной подготовки студентов: Матер. научно-методической конференции/ Уральский гос. пед. ун-т. Екатеринбург, 1995. С. 66-67.
50. Горский Д.П. Обобщение и познание. М.: Мысль, 1985. 208 с.

51. Грабарь М.И. Некоторые положения выборочного метода в связи с организацией изучения знаний учащихся. - М.: Педагогика, 1973. 46 с.
52. Грабарь М.И. Применение математической статистики в педагогических исследованиях. - М.: Педагогика, 1977. 136 с.
53. Грабарь М.И., Краснянская К.А. Применение математической статистики в педагогических исследованиях: непараметрические методы. М.: Педагогика, 1977. 120 с.
54. Громыко Ю.В. Проектирование и программирование развития образования. Московский учебник. 1996. 545 с.
55. Громыко Ю.В. Роль взаимопонимания при решении учебных задач в совместной деятельности Автореф. дис. ... канд. психол. наук. М, 1985. 25 с.
56. Давыдов В. В. О месте категории деятельности в современной теоретической психологии // Деятельность: теории, методология, проблемы - М.: Политиздат, 1990. С.143-156.
57. Давыдов В.В. Виды обобщения в обучении: Логико-психологические проблемы построения учебных предметов. - М.: Педагогика, 1972. 424 с.
58. Давыдов В.В. Виды обобщения в обучения. - М.: Педагогика, 1972. 424 с.
59. Давыдов В.В. Проблемы развивающего обучения: Опыт теоретического и экспериментального исследования. - М.: Педагогика, 1986. 388 с.
60. Давыдов В.В. Теория развивающего обучения. М.: Педагогика, 1996. 544 с.
61. Демакова И., Годунова Е., Борисова В., Жамкочьян М., Матвеева Т., Олейник М., Петрова И., Пронина Е., Пятунин В. На подступах к компьютерной педагогике. // Информатика и образование, 1990. № 6. С. 82-87.

62. Демушкин А.С., Кириллов А.И., Сливина Н.А., Чубров Е.В., Кривошеев А.О., Фомин С.С. Компьютерные обучающие программы // Информатика и образование, 1995. № 3. С. 15-22.
63. Джакупов С.М. Психологическая структура процесса обучения: Дисс. ... докт психол. наук. - М., 1998. 355 с.
64. Джонсон Дж.К. Методы проектирования. М.: Прогресс, 1986. 326 с.
65. Дидактика технологического образования: Книга для учителя. Часть 2 / Под ред. П.Р. Атутова. - М.: ИОСО РАО, 1998. 234 с.
66. Дизайн и проектная наука: Теорет. курс / Моск. экстер. гуманитар ун-т. - М.: Акад. изд-во МЭГУ, 1997. 157 с.
67. Диксон Дж. Проектирование систем: изобретательство, анализ и принятие решений. - М.: Мир, 1969. 435 с.
68. Дитрих Я. Проектирование и конструирование: системный подход. - М.: Мир, 1981. 456 с.
69. Дьюи Д. Психология и педагогика мышления. Пер. с англ. Н.М. Никольской. - М.: Совершенство, 1997. 208 с.
70. Дьюи Дж. Школа будущего - М.: Госиздат, 1926.
71. Дэвид Х. Джонсон Компьютеры как инструменты познания: изучение с помощью технологии, а не технологии // Информатика и образование, 1996. № 4. С. 117-131.
72. Емельянов Ю.Н. Активное социально-психологическое обучение - Л.: ЛГУ, 1985. 167 с.
73. Журавлев И.К. Зорина Л.Я., Лернер И.Я. Дидактическая концепция содержания базового содержания образования // Дидактические проблемы построения базового содержания образования. - М.: ИТПиМИО РАО, 1993. С. 162-203.
74. Завалишина Д.Н., Ломов Б.Ф., Рубахин В.Ф. Уровни и этапы принятия решений // Проблемы принятия решений. - М.: Наука, 1976. 312 с.

75. Здравомыслов А.Г. Потребности: Интересы: Ценности. - М., Политиздат, 1986. 223 с.
76. Зейгарник Б.В. Теории личности в зарубежной психологии. - М., Изд-во МГУ, 1982. 128 с.
77. Злобин Н.С. Терминологическая разногласица и (или) концептуальные расхождения: Деятельность: теории, методология, проблемы. - М.: Педагогика, 1990. 122 с.
78. Зуев П.В. Теоретические основы эффективного обучения физике в средней школе (праксеологический подход): Монография / Урал. гос. пед. ун-т. - Екатеринбург, 2000. 153 с.
79. Информатика: энциклопедический словарь для начинающих / сост. Поспелов Д.А. - М.: Педагогика-Пресс, 1994. 352 с.
80. Каптерев П.Ф. Избранные педагогические сочинения. - М.: Педагогика, 1982.
81. Карпов А.В. Процессы принятия решения в регуляции деятельности // Психологический журнал, 1991. № 1. С. 12-21.
82. Карпович В.Н. Системность теоретического знания: логический аспект. - Новосибирск: Наука, 1984. 125 с.
83. Качество знаний учащихся и пути его совершенствования / Под ред. М.Н. Скаткина, В.В. Краевского. - М.: Педагогика, 1978. 208 с.
84. Кивчун С.Н. Особенности мотивации учебной трудовой деятельности младших школьников в различных условиях обучения: Дисс. ... канд. психол. наук - М., 1984. - 182 с.
85. Килпатрик У.Х. Метод проектов: Применение целевой установки в педагогическом процессе. - Л.: Брокгауз-Ефрон, 1925. - С. 10-12.
86. Кларин М.В. Инновационные модели обучения в зарубежных педагогических поисках. - М.: Арена, 1994 - 222 с.

87. Климов Е.А. Индивидуальный стиль деятельности в зависимости от типологических свойств нервной системы / Казанский гос. ун-т, Казань, 1969. 278 с.

88. Кодухов В.И. Интенсификация и оптимизация учебной деятельности. // Оптимизация и интенсификация педагогического процесса в школе и вузе. - Ишим. - 1994. С. 8-12.

89. Коллингс Е. Опыт работы американской школы по методу проектов. - М.: Новая Москва, 1926. 62 с.

90. Коменский Я.А. Избранные педагогические сочинения. Т1. Великая дидактика. М.: Педагогика, 1982. 656 с, ил.

91. Компьютеры в обучении: шведский путь. Перевод Каптелинин В.Н.//Информатика и образование, 1992. № 1. С. 112.

92. Компьютеры, модели, вычислительный эксперимент: Введение в информатику с позиций математического моделирования / Авт. пред. Самарский А.А. - М.: Наука, 1988. 176 с.

93. Коуров В.Г. Адаптивные автоматизированные среды обучения на основе экспертных технологий. Дисс. ... докт. техн. наук. Новосибирск, 1994. 226 с.

94. Коуров В.Г., Осипов Н.А. Инструментальные средства искусственного интеллекта. - Новосибирск-Шадринск, 1993. 156 с.

95. Коуров В.Г., Пирогов В.Ю., Пирогова Т.Ф., Беспалько И.И., Рязанов А.В., Слинкин Д.А. Основы информатики: Методическое пособие для абитуриентов/ Изд-во Шадр. гос. пед. ин-та. Шадринск, 1995. 50 с.

96. Краснова О.И., Левченко И.В. Излучение компьютера и здоровье детей. // Информатика и образование, 1995. № 1. С. 113-115.

97. Ксензова Г.Ю. Перспективные школьные технологии: Учебно-методическое пособие. - М.: Педагогическое общество России, 2000. - 224 с.

98. Куделин Е. Е. Диалектика производства потребностей. - М.: Политиздат, 1977. 175 с
99. Кудрявцев А.В., Слинкин Д.А. Рекурсивные алгоритмы в задачах и играх. Учебное пособие для студентов вузов/ Шадринский пед. ин-т. Шадринск, 1999. 30 с.
100. Ланина И.Я. Формирование познавательных интересов учащихся на уроках физики. - М.: Просвещение, 1985. 128 с.
101. Лаутербах Р., Фрей К. Программное обеспечение процесса обучения // Перспективы: Вопросы образования, 1988. № 3. С 70-79.
102. Леонтьев А.Н. Деятельность. Сознание. Личность. - М.: 1975. 304с.
103. Леонтьев А.Н. Избранные психологические произведения. В 2-х т. Т.1 / Под ред. Давыдова В.В. и др. - М.: Педагогика, 1983. 131 с.
104. Лернер И.Я. Дидактические основы методов обучения. - М.: Педагогика, 1983. 178с.
105. Лихачев Б.Т. Педагогика. Курс лекций: Учебное пособие для студентов. - М, Прометей, 1996. 528 с.
106. Матрос Д.Ш., Полев Д.М., Мельникова Н.Н. Управление качеством образования на основе новых информационных технологий и образовательного мониторинга. Издание 2-е, исправленное и дополненное. - М.: Педагогическое общество России, 2001. - 128 с.
107. Матяш Н.В. Проектный метод обучения в системе технологического образования // Педагогика, 2000. № 4. С.38
108. Матяш Н.В. Психология проектной деятельности школьников в условиях технологического образования / Под ред. В.В. Рубцова. - Мозырь: РИФ "Белый ветер", 2000.-286с
109. Машбиц Е.И. Диалог в обучающей системе. - Киев.: Высшая школа, 1989. 182 с.

110. Машбиц Е.И. Психолого-педагогические проблемы компьютеризации обучения. - М.: Педагогика, 1988. 191 с.
111. Методика педагогического исследования и некоторые показатели эффективности используемых средств обучения. Методические указания. /Под. ред. Гресс. Н.П. - Челябинск: Изд-во Челябинского политехн. ин-та им. Ленинского комсомола, 1982. 65с.
112. Минина Е.Е. Статистические методы в педагогических экспериментах: Методические рекомендации. - Екатеринбург: УНФ "Наука", 1995. 38 с.
113. Могилев А.В., Хеннер Е.К. О понятии "Информационное моделирование" // Информатика и образование, 1997. № 8. С. 3-7.
114. Монахов В.М. Технологические основы проектирования и конструирования учебного процесса. - Волгоград, 1995. 234 с.
115. Монахов В.М. Что такое новая информационная технология? // Математика в школе. 1990. № 2. С. 34-39.
116. Нарыкова И. Компьютерное моделирование в Великобритании // Информатика и образование, 1992. № 3-4. С.116 - 120.
117. Невуева Л., Сергеева Т. О перспективных тенденциях разработки педагогических программных средств // Информатика и образование, 1990. № 5. С. 5-10.
118. Немов Р.С. Психология. Учебник для высших педагогических учебных заведений в 3-х книгах. Книга 1. Общие основы психологии. - М.: Просвещение, 1995. 576 с.
119. Нечаев Н.Н. Проектное моделирование как творческая деятельность. Дис. ... докт. псих. наук. - М., 1987. 411с.
120. Николов Р.В., Бычваров С.Х. Автоматизированная среда для обучения информатике // Персональные ЭВМ в задачах информатики / Под ред. Ершова А.П. - Новосибирск, 1984. С 105-116.

121. Новые педагогические и информационные технологии в системе образования: Учеб. пособие для студ. пед. вузов и системы повыш. квалиф. пед. кадров / Е.С.Полат, М.Ю.Бухаркина, М.В.Моисеева, А.Е.Петров; Под ред. Е.С. Полат. – М.: Издательский центр "Академия", 2000. – 272 с.
122. Ожегов С.И. Словарь русского языка /Под ред. чл.-корр. АН СССР Н.Ю. Швецовой, - М.: Русский язык, 1987. 750с.
123. Около 100 российских обучающих и образовательных CD-ROM. // Компьютер-пресс, 1998. № 9. С.77-93.
124. Организация учебной деятельности в терминальном классе на базе Микро-ЭВМ: Учеб.пособие. - Новосибирск, 1989. 98 с.
125. Павлова М.Б. О проектном подходе к разработке содержания нового учебного предмета "Технология" // Школа и производство, 1993. № 5. -С.43-45.
126. Паршин В.М. Опыт работы в лаборатории "Введение в технологию". // Информатика и образование, 1997. № 1. С. 110-115.
127. Паск Г. Обучение как процесс создания системы управления. // Кибернетика и проблемы обучения /Под ред. Берга А.И. - М: Прогресс, 1970. С 25-86.
128. Педагогика: Учеб. пособие для студентов пед. ин-тов. / Под ред. Бабанского Ю.К. - М.: Просвещение, 1988. 479 с.
129. ПЕДАГОГИКА: Учебное пособие для студентов педагогических учебных заведений / В.А. Слостенин, И.Ф. Исаев, А.И.Мищенко, Е.Н.Шиянов. – М.: Школа-Пресс, 1998. – 512 с.
130. Пейперт С. Переворот в сознании: дети, компьютеры и плодотворные идеи: Пер. с англ. / Под ред. Беляевой А.В., Леонаса В.В. - М.: Педагогика, 1989. - 224с.

131. Пиаже Ж. Избранные психологические труды: Психология интеллекта: Генезис числа и ребенка: Логика и психология. - М.: Просвещение, 1969. 660 с.
132. Полат Е.С. Компьютер и школа. // Физика в школе, 1985. № 2. С. 51-55.
133. Полат Е.С. Метод проектов на уроках иностранного языка. <http://www.ioso.ru/distant/iaproj.htm>
134. Психологическая диагностика и новые информационные технологии. - СПб., 1997. 207 с.
135. Психологический словарь /под ред. Давыдова В.В. - М.: Педагогика, 1983. 448 с.
136. Психология: Словарь / Под ред. А.В. Петровского, М.Г. Ярошевского - М.: Политиздат, 1990г. 494 с.
137. Пустильник И.Г. Теоретические основы формирования научных понятий у учащихся /Уральский гос. пед. ун-т, Екатеринбург, 1997. 90 с.
138. Пушкин В.Г., Урсул А.Д. Информатика, кибернетика, интеллект: Философские очерки. - Киченев: Штинница, 1989. 278 с.
139. Разумовский В.. Обучение и научное познание // Педагогика. 1997. № 1. С. 10-16.
140. Роберт И. Новые информационные технологии в обучении: дидактические проблемы, перспективы использования // Информатика и образование, 1991. № 4. С. 18-25.
141. Рогов Е.И. Психология человека. - М.: Гуманит. Изд. Центр ВЛАДОС, 1999. 320 с
142. Романов А.В. Совершенствование профессиональной подготовки учащихся школ физико-математического профиля к использованию средств информационных технологий: Дис. ... канд.пед.наук. - Тамбов, 1997. - 223с.

143. Российская педагогическая энциклопедия: В 2 т. / Гл ред. В. В. Давыдов. - М.: Большая Российская энциклопедия, 1993. 608 с.
144. Российские образовательные CD-ROM: сегодня и вчера. // Компьютер-пресс, 1998. № 9. С. 74-76.
145. Сапрыкина Г.А. Педагогические программные средства для индивидуализации школьного образовательного процесса: Дис. ... канд. пед. наук. – Новосибирск, 1997. 165с.
146. Селевко Г.К. Современные образовательные технологии. – М.: Народное образование, 1998. 256 с.
147. Скаткин М.Н. Методология и методика педагогических исследований. - М.: Педагогика, 1986. 150 с.
148. Скок Г.Б. Как проанализировать собственную педагогическую деятельность: Учеб. пособие для преподавателей / Отв.ред. Ю.А. Кудрявцев. - М.: Педагогическое общество России, 2000. – 102 с.
149. Слинкин Д.А. Анализ понятия операционализма в психологии познания с точки зрения основных гносеологических принципов // Естественные науки в педвузе и школе: Матер. межвуз. науч.-методич. конференции/ Уральский гос. пед. ин-т., Екатеринбург 1995. С. 51-53.
150. Слинкин Д.А. Использование локальной сети для постановки компьютерных демонстраций // Инновационные процессы в подготовке будущего учителя физики: Матер. XXIX зонального совещания преподавателей педвузов Урала, Сибири и Дальнего Востока 14-17 окт. 1996 г.: В 3ч. Ч 2 / Урал. гос. пед. ун-т. Екатеринбург, 1996. С. 63-64.
151. Слинкин Д.А. Новые методы в программируемом обучении // Развитие творческой активности учащихся в процессе обучения и профессиональной подготовки студентов: Матер. научно-методической конференции/ Уральский гос. пед. ун-т. Екатеринбург, 1995. С. 70-72.
152. Слинкин Д.А. Об одной реализации "Компьютерной физической лаборатории"// Межпредметные связи информатики с естественно-

математическими дисциплинами: Тезисы научно-практической конференции/ Курганский гос. ун-т. Курган, 1994. С. 14-15.

153. Слинкин Д.А. Обучение программированию на основе метода проектов в старших классах общеобразовательной школы // Повышение эффективности подготовки учителей физики, информатики, технологии в условиях новой образовательной парадигмы: Тезисы научно-практической конференции, г. Екатеринбург, 21-22 апреля 2001 г. / Уральский гос. пед. ун-т. Екатеринбург, 2001.

154. Слинкин Д.А. Обучение программированию на основе метода проектов в старших классах общеобразовательной школы // Повышение эффективности подготовки учителей физики, информатики, технологии в условиях новой образовательной парадигмы: Тезисы научно-практической конференции, г. Екатеринбург, 21-22 апреля 2001 г. Ч.2. / Уральский гос. пед. ун-т. Екатеринбург, 2001. С. 11-13.

155. Слинкин Д.А. Организация факультативного обучения программированию в педагогическом вузе // Информатизация образования 2001: Тезисы научно-практической конференции, г. Екатеринбург, 13-16 февраля 2001 г. / Уральский гос. пед. ун-т. Екатеринбург, 2001. С. 239-240.

156. Слинкин Д.А. Особенности организации "метода проектов" при изучении языков и методов программирования в педагогическом вузе // Информатизация образования 99: Тезисы научно-практической конференции, г. Шадринск, 22-24 марта 1999 г. / Шадринский гос. пед. ин-т. Шадринск, 1999. С. 8-9.

157. Слинкин Д.А. Программирование. Часть 1. Язык программирования Турбо-Паскаль: Учебное пособие для студентов вузов / Шадринский гос. пед. ин-т. Шадринск, 1999. – 140 с.

158. Слинкин Д.А. Программирование. Часть 2. Методы программирования на Турбо-Паскале: Учебное пособие для студентов вузов / Шадринский гос. пед. ин-т. Шадринск, 2000. – 136 с.

159. Слинкин Д.А., Грязева Н.Н. Использование КМС при решении задач с критическими данными // Информатизация образования 99: Тезисы научно-практической конференции, г. Шадринск, 22-24 марта 1999 г. / Шадринский гос. пед. ин-т. Шадринск, 1999. С. 60-61.
160. Слинкин Д.А., Шамало Т.Н. Роль метода проектов в развитии познавательных потребностей школьников и студентов педагогического вуза // Информатизация образования 2001: Тезисы научно-практической конференции, Екатеринбург, 13-16 февраля 2001 г. / Уральский гос. пед. ун-т. Екатеринбург, 2001. С. 240-241.
161. Слинкин Д.А., Шохирев М.В. Особенности реализации Web-сервера в Intranet - сети педагогического вуза // Информатизация образования 99: Тезисы научно-практической конференции, г. Шадринск, 22-24 марта 1999 г. / Шадринский гос. пед. ин-т. Шадринск, 1999. С. 26-27.
162. Современная западная философия: словарь / Сост.: Малахов В.С., Филатов В.П. – М.: Политиздат, 1991. 414 с.
163. Современная философия: Словарь и хрестоматия. - Ростов-на-Дону: Феникс, 1996. 511 с.
164. Спиркин А.Г. Философия: Учебник. – М.: Гардарики, 1999. 816 с.
165. Сьюэлл Дейв Ф., Ротерей Дейвид Р. Основные направления применения ЭВМ // Перспективы: Вопросы образования, 1988. № 3. С. 60-69.
166. Тихомиров О.К. Психология мышления: Учебное пособие. М.: Изд-во Моск. ун-та, 1984. 272 с.
167. Уледов А.К. Духовная жизнь общества: Пробл. методол. исслед. – М.: Мысль, 1980. – 271 с.
168. Философский словарь / Под ред. Фроловой И.Е. - М.: Политиздат, 1991. - 560с.

169. Философский энциклопедический словарь. - М.: ИНФР, 1997. 576с.
170. Френд Д. Интеграция вычислительной техники в школы // Перспективы: Вопросы образования, 1988. № 3. С. 46-59.
171. Фридман Л.М. Наглядность и моделирование в обучении. М.: Знание, 1984. 80 с, ил.
172. Фролова Г.В. Педагогические возможности ЭВМ. - Новосибирск: Наука. Сиб. отд-ние, 1988. 173 с.
173. Хантер Б. Мои ученики работают на компьютерах: Пер. с англ. М.: Просвещение, 1989. 224 с.
174. Чошанов М.А. Дидактическое конструирование гибкой технологии обучения // Педагогика, 1997. № 2. С. 25-27.
175. Шалютин С.М. Абстрактное мышление и кибернетика / Челябинский гос. пед. ин-т, Челябинск, 1976, 108 с.
176. Шамало Т.Н. Теоретические аспекты проблемы развития творческого мышления в учебном процессе // Развитие творческой активности учащихся в процессе обучения и профессиональной подготовки студентов: Матер. научно-методической конференции/ Уральский гос. пед. ун-т. Екатеринбург, 1995. С. 3-5.
177. Шамало Т.Н. Теоретические основы использования физического эксперимента в развивающем обучении. Свердловск 1990. 96 с.
178. Шамало Т.Н., Слинкин Д.А. Использование персонального компьютера в обучении студентов технике физического эксперимента// Гуманизация педагогического образования: Тезисы научно-методической конференции/ Уральский гос. пед. ин-т. Екатеринбург 1994. С. 71-72.
179. Шамало Т.Н., Слинкин Д.А. Реализация функций дидактического процесса в средней школе средствами педагогических программных средств // Информатизация образования 99: Тезисы научно-практической

конференции, г. Шадринск, 22-24 марта 1999 г. / Шадринский гос. пед. ин-т. Шадринск, 1999. С. 15-16.

180. Шишов С.Е., Кальней В.А. Школа: мониторинг качества образования. - М.: Педагогическое общество России, 2000. - 320 с., приложения.

181. Anderson, J.R. The Architecture of Cognition, Harvard University Press Cambridge, MA. 1983.

182. Barron D.W. An introduction to the study of programming languages. Cambridge University Press. 1977.

183. Bloom, B. (ed) Taxonomy of Educational Objectives - Handbook I: Cognitive Domain, David McKay Co, New York. 1956.

184. Gagne, R. and Briggs, L. Principles of Instructional Design, (2nd ed.), Holt, Rinehart and Winston, New York. 1979.

**Анкета для преподавателей информатики педагогического
института и учителей школ**

Уважаемые преподаватели и учителя! Проблема построения процесса обучения языкам и методам программирования является одной из актуальных в современных социально-экономических условиях.

В целях совершенствования процесса обучения учащихся и студентов просим Вас принять участие в исследовании и ответить на несколько вопросов, касающихся названной проблемы. Надеемся, что Ваши ответы будут заинтересованными и объективными. Возможные варианты ответов на некоторые ответы напечатаны на бланке. Если они выражают Ваше мнение, подчеркните их.

1. Фамилия, имя, отчество _____

2. Вы работаете:

в школе учителем информатики;

преподавателем ВУЗа;

в школе и в ВУЗе.

3. Название учебного заведения _____

4. Какое учебное заведение и в каком году закончили? _____

5. Ваш стаж работы в школе и в ВУЗе _____

6. Ваше ученое звание, ученая степень, должность _____

7. Ваша категория (для учителей школ) _____

8. Считаете ли Вы себя в программировании специалистом:

начинающим;

- среднего уровня;
- высокого уровня;
- компетентным;
- мастером своего дела.

9. На каком факультете и курсе или в каких классах Вы преподаете информатику? _____

10. Какие языки программирования Вы используете или преподаете?

11. Что в организации процесса обучения Вас больше всего удовлетворяет, а что нет? _____

12. Какие затруднения испытываете? _____

13. Что для ученика или студента интереснее:

учиться пользоваться готовым программным обеспечением в виде работы с системой встроенной помощи и по объяснениям учителя;

самостоятельно постигать азы профессиональной компетентности оператора ЭВМ (пользовательский уровень владения компьютерной техникой);

решать задачи, сформулированные в терминах информатики. Примеры: 1) отсортировать массив...; 2) найти из трех чисел максимальное...; 3) вычислить факториал числа... и т.п.;

решать задачи, имеющие занимательную формулировку. Пример: 1) занятия в начальных классах отменяются в тех случаях, если температура воздуха не ниже 25 градусов мороза, а также при ветре не менее 7 м/с и температуре не ниже 20 градусов мороза. По утренней сводке определить, пойдет ли дети в школу; 2) определить по возрасту человека долго ли ему осталось до пенсии. Если он пенсионер, поздравить его; 3) по чис-

лу, означающему месяц рождения (например, январь – 1, февраль – 2 и т.д.)
определить время года;

- придумывать самостоятельно задачи и затем решать их;
- создавать небольшие программные комплексы прикладного характера;
- создавать большие программные комплексы, на написание которых предназначена целая четверть, полугодие, год;
- участие в олимпиадах, конкурсах, выставках;
- участие в научных объединениях.

Ваши комментарии к ответу _____

14. Подготовка каких форм дается наиболее легко, каких трудно? ____

15. В каких формах Вы предпочитаете объединять усилия школьников или студентов? _____

16. Что является наиболее эффективным при изучении языка программирования:

- изучение языка блоками с зачетом по окончании изучения каждого;
- решение большого количества разнообразных задач по программированию;
- создание большого программного проекта, включающего в создание использование всех знаний по программированию, изучение которых запланировано на семестр (четверть, полугодие, год);
- создание серии программных проектов прикладного характера;

создание систем задач по программированию, направленных на изучение отдельных методов и языковых конструкций языка.

17. Какими, по Вашему мнению, знаниями и умениями из перечисленных должен обладать будущий учитель информатики:

знание языка программирования (укажите каких) _____;

знание численных методов программирования;

знание основных нечисловых алгоритмов (например, поиск и сортировка);

знание основных методов создания интерфейса программного комплекса (например, создание "рабочего поля", меню, кнопок, полей ввода данных и т.д.);

знание методов создания простейших трансляторов и компиляторов;

знание основных требований к программным комплексам учебного и профессионального назначения (например, к обучающим программам для младших школьников, к помощи для программного обеспечения ЭВМ; программам бухгалтерского направления и т.д.);

умение планировать работу студентов или школьников по изучению того или иного раздела программирования;

умение прогнозировать результаты работы студентов;

умения осуществлять процесс обучения;

умение анализировать и применять новые методы обучения;

умение показать новое в знакомом процессе решения задачи, создания программного комплекса и т.д.;

умение систематизировать знания полученные студентами или школьниками;

учение составлять схемы, графики, таблицы систематизирующего характера.

18. Выделите основной метод, используемый Вами при обучении информатике:

- метод проектов;
 - метод решения разнообразных задач;
 - метод промежуточных зачетов;
 - метод решения систем задач;
 - Ваш метод _____
-

19. Используйте ли Вы в процессе обучения метод проектов:

- да, это основной метод обучения;
- да, но его использование зависит от темы, предмета, особенностей класса;
- очень редко, для небольшого числа учащихся;
- не использую.

20. Выделите те формы работы, которые по Вашему мнению наиболее популярны среди учащихся:

- изучение ПО;
- создание больших программных комплексов;
- решение задач;
- участие в конкурсах;
- участие в олимпиадах.

Благодарим за помощь!

**Схема-тест уровня знаний по программированию у
учителей и преподавателей**

Перед Вами бланк, имеющий вид таблицы. В нем 36 пронумерованных задач. Для каждой задачи проделайте следующие действия

1. Решите задачу.

2. Определите, к каким разделам изучения алгоритмизации, программирования или языка программирования можно отнести данную задачу.

Пользуйтесь следующими обозначениями:

ЛП – программирование линейных алгоритмов;

ПВ – программирование ветвлений;

ПЦ – программирование циклов;

М – изучение массивов;

СП – изучение строковых переменных;

ПФ – изучение процедур, функций.

3. Определите приоритет (от 1 до 36) решения данной задачи на уроках информатики по отношению к другим предложенным задачам.

Благодарим за помощь!

№1. Даны два числа. Найти их сумму, разность и произведение	№2. Даны три числа. Определить, существует ли треугольник с такими сторонами	№3. Вывести на экран таблицу умножения	№4. Найти сумму и произведение всех элементов массива	№5. Удвоить каждый символ строки	№6. Создать функцию, которая по значению N возвращает сумму элементов столбца N
№7. Даны три целых числа. Возвести в куб то из них, которое больше 0.	№8. Вывести на экран первые N положительных четных чисел в обратном порядке	№9. Заменить нулями все максимальные элементы массива	№10. Обрезать строку за последним пробелом	№11. Создать процедуру, которая дописывает переданную строку пробелами до длины строки N	№12. Даны два числа a и b. Найти радиус кольца (радиусы окружностей - a и b)
№13. Вывести на экран первые N чисел/сел/	№14. Увеличить каждый элемент массива на шаг	№15. Дано натуральное число. Вывести значение его квадрата в прямом и обратном порядке	№16. Создать функцию, которая возвращает сумму элементов главной диагонали массива, если $a[1,1]=1$	№17. Вычислить длину отрезка по координатам его концов	№18. Определить, в каком квадранте находится точка с координатами x, y.
№19. Найти индекс и значение последнего минимального элемента одномерного массива	№20. Вставить пробел после каждой запятой в строке.	№21. Создать функцию, которая возвращает 1, если в одномерном массиве все числа одинаковы	№22. Дан первый член геометрической прогрессии и частное. Найти сумму первых n членов	№23. Найти корни квадратного уравнения	№24. По алгоритму Евклида найти НОД и НОК двух чисел a и b.
№25. Заменить в строке все знаки присваивания "=" на "=="	№26. Создать процедуру, которая в переданной ей строке все точки заменит на !	№27. Дано ребро куба. Найти площадь боковых граней	№28. Дано целое число N. Если $N \in [2,7]$, то $y=2N$. Если $N \in [-10,1]$, то $y=3$. В противном случае $y=0$	№29. Найти произведение цифр натурального числа	№30. Перенести содержание двумерного массива в одномерный
№31. Создать функцию, которая возвращает 1, если переданное ей число простое	№32. Дано число a. Найти a^{64} за 4 операции	№33. Даны три числа. Вывести на экран их в порядке возрастания.	№34. Определить, является ли число совершенным	№35. Найти максимальный и минимальный элемент двумерного массива	№36. "Перевернуть" строку

**Анкета для учащихся старших классов общеобразовательных
школ и студентов педагогических вузов**

Уважаемые школьники и студенты! Проблема построения процесса обучения языкам и методам программирования является одной из актуальных в современных социально-экономических условиях.

В целях совершенствования процесса обучения просим Вас принять участие в исследовании и ответить на несколько вопросов, касающихся названной проблемы. Надеемся, что Ваши ответы будут заинтересованными и объективными. Возможные варианты ответов на некоторые вопросы напечатаны на бланке. Если они выражают Ваше мнение, подчеркните их.

1. Фамилия, имя _____
2. Школа (факультет) _____
3. Класс (группа) _____
4. С какого времени Вы занимаетесь информатикой?
 - до школы;
 - в начальной школе;
 - в среднем звене общеобразовательной школы;
 - в старшем школьном возрасте;
 - после поступления в вуз.
5. Вам это интересно?
 - это мое хобби;
 - интересно;
 - так себе;
 - совсем не интересно.

6. С какого класса Вы начали изучение программирования?

7. Введен ли у Вас в школе метод проектов для обучения информатике?

да;

нет.

Следующий блок вопросов предназначен тем, кто с методом проектов уже сталкивался в обучении информатике.

8. Как давно Вы работаете по этому методу? _____

9. Сколько проектов Вами было уже написано? _____

10. Какие языки программирования для этого Вы использовали? _____

11. Нравятся ли Вам проекты, которые Вы сделали сами? _____

12. Выделите несколько наиболее удачных Ваших проектов _____

13. Ваше мнение, насколько оправдано применение этой методики для обучения программированию? _____

14. Выделите, пожалуйста, что Вам в этом методе нравится? _____

15. По вашему мнению, какие недостатки у него имеются? _____

16. Ваши пожелания и предложения по поводу улучшения эффективности применения этой методики в процессе обучения программированию.

Следующий блок вопросов предназначен для тех школьников и студентов, которые не сталкивались с этим методом в процессе обучения.

Метод проектов подразумевает написание в процессе обучения (всего курса, по четвертям или полугодиям) больших программных комплексов (например, графический или текстовый редактор, программа для автоматизации работы мед. сестер в регистратуре поликлиники и т.д.).

17. Считаете ли Вы необходимым обучение написанию таких программных комплексов в школе или в вузе? _____

18. Интересно ли лично Вам было бы этим заниматься? _____

19. Нравится ли Вам, как в настоящее время Вы изучаете программирование? _____

20. В чем Вы видите достоинства той методики, по которой работает сейчас Ваш учитель? _____

21. что Вам бы хотелось изменить? _____

Спасибо Вам за помощь!