Ilmiy tadqiqotlar va ularning yechimlari jurnali
Aoka guvohnoma № 323759 ISSN: 3060-4818

РАЗВИТИЕ КРИТИЧЕСКОГО И АЛГОРИТМИЧЕСКОГО МЫШЛЕНИЯ УЧАЩИХСЯ НА УРОКАХ ИНФОРМАТИКИ НА ОСНОВЕ ПРОЕКТНО-ИССЛЕДОВАТЕЛЬСКОГО ОБУЧЕНИЯ

Очилова Дилсуз Акрамовна
исследователь Каршинского государственного университета
https://doi.org/10.5281/zenodo.22989031

Аннотация. В статье исследуются педагогические возможности проектно-исследовательского обучения в развитии критического и алгоритмического мышления учащихся на уроках информатики. Раскрыты теоретические основы интеграции проектной и исследовательской деятельности, определены этапы организации учебных проектов, предложена система критериев и показателей оценки мыслительной самостоятельности учащихся. Особое внимание уделено постановке проблемных задач, разработке алгоритмов, поиску и проверке информации, выбору цифровых средств, тестированию решений и рефлексии. Разработана методическая модель, включающая мотивационно-проблемный, аналитико-поисковый, алгоритмический, практико-технологический, презентационно-оценочный и рефлексивный этапы. Показано, что сочетание проектного и исследовательского подходов позволяет формировать у учащихся способность самостоятельно выявлять проблему, выдвигать гипотезы, строить последовательность действий, сравнивать альтернативы, аргументировать решения и корректировать ошибки. Предложены примеры проектов для 7–11 классов и комплексная рубрика оценивания.

Ключевые слова: информатика, проектно-исследовательское обучение, критическое мышление, алгоритмическое мышление, самостоятельная деятельность, цифровая компетентность, проблемное обучение, рефлексия, оценивание.

ВВЕДЕНИЕ

Современное школьное образование развивается в условиях интенсивной цифровизации, быстрого обновления информации и расширения сферы применения алгоритмов и интеллектуальных технологий. В этих условиях результат обучения не может ограничиваться воспроизведением готовых знаний. Учащийся должен уметь самостоятельно обнаруживать проблему, анализировать исходные данные, проверять достоверность информации, выбирать способ решения, строить последовательность действий и оценивать полученный результат. Именно поэтому развитие критического и алгоритмического мышления становится одной из центральных задач преподавания информатики.

Информатика занимает особое место в формировании указанных качеств. Содержание предмета непосредственно связано с моделированием, алгоритмизацией, программированием, обработкой данных, поиском информации и использованием цифровых инструментов. Однако при преимущественно объяснительно-иллюстративной организации урока потенциальные возможности предмета реализуются не полностью. Если учащийся получает не только инструкцию, но и проблемную ситуацию, требующую самостоятельного поиска, сравнения

вариантов и создания конкретного цифрового продукта, учебная деятельность приобретает исследовательский характер.

Проектно-исследовательское обучение представляет собой интеграцию двух взаимодополняющих подходов. Проектная составляющая ориентирована на получение практического результата — программы, сайта, базы данных, цифровой модели, презентации, информационного ресурса или автоматизированного решения. Исследовательская составляющая обеспечивает постановку вопроса, формулирование гипотезы, сбор и анализ данных, проверку вариантов, объяснение причин ошибок и аргументацию выводов. Такое соединение особенно продуктивно на уроках информатики, поскольку позволяет связать знания с реальными задачами и одновременно развивать мыслительные операции высокого уровня.

Цель статьи состоит в теоретическом обосновании и методической разработке модели развития критического и алгоритмического мышления учащихся на уроках информатики средствами проектно-исследовательского обучения. Для достижения цели решаются следующие задачи: уточняется педагогическое содержание критического и алгоритмического мышления; раскрываются возможности проектно-исследовательского подхода; предлагаются этапы организации учебной деятельности; разрабатываются критерии оценки; формулируются практические рекомендации для учителя.

ОБЗОР ЛИТЕРАТУРЫ И ТЕОРЕТИЧЕСКИЕ ОСНОВЫ

Идея обучения через деятельность имеет глубокие педагогические основания. Дж. Дьюи связывал полноценное усвоение знаний с опытом учащегося и решением практически значимых задач [1]. У. Килпатрик развил данный подход в «методе проектов», рассматривая проект как целенаправленную деятельность, объединяющую замысел, самостоятельное действие и конкретный результат [2]. В дальнейшем проектное обучение получило развитие как технология, в которой центральное место занимают проблема, самостоятельный поиск, сотрудничество и создание продукта.

В исследованиях P. C. Blumenfeld и соавторов показано, что проектная деятельность повышает учебную мотивацию, если задача является содержательно значимой, допускает выбор и требует применения знаний в процессе создания результата [3]. J. W. Thomas подчеркивает, что полноценное проектное обучение предполагает центральность проекта для содержания курса, конструктивное исследование, автономию учащегося и реалистичность задачи [4]. S. Bell связывает проектное обучение с формированием навыков XXI века, в частности сотрудничества, коммуникации, самостоятельного поиска и решения проблем [5].

Для информатики принципиальное значение имеет понятие алгоритмического и вычислительного мышления. J. Wing определяет computational thinking как способ постановки проблем и построения решений в форме, пригодной для эффективного выполнения человеком или вычислительной системой [6]. В школьной практике это проявляется в декомпозиции задачи, выделении существенных данных, построении алгоритма, поиске закономерностей, тестировании и отладке. Идеи S. Papert о конструкционизме также подтверждают значение самостоятельного создания цифровых объектов для глубокого освоения понятий и развития мышления [7].

Критическое мышление в контексте информатики включает способность оценивать надежность цифровых источников, различать факт и мнение, проверять корректность данных, обнаруживать логические ошибки, сравнивать альтернативные способы решения и аргументировать выбор. Алгоритмическое мышление, в свою очередь, обеспечивает структурирование действий, последовательность операций, выявление условий и повторений, построение модели решения и оценку эффективности алгоритма. Эти два типа мышления не противопоставляются друг другу: критический анализ помогает выбрать и обосновать алгоритм, а алгоритмическая организация делает рассуждение последовательным и проверяемым.

Таким образом, проектно-исследовательское обучение можно рассматривать как педагогическую среду, в которой критическое и алгоритмическое мышление формируются не изолированно, а в едином цикле: проблема — гипотеза — информация — алгоритм — реализация — проверка — аргументация — рефлексия.

МЕТОДОЛОГИЯ ИССЛЕДОВАНИЯ

Методологическую основу исследования составили системный, деятельностный, компетентностный и проблемно-исследовательский подходы. Использовались методы анализа научно-педагогической литературы, сравнительного анализа, педагогического моделирования, наблюдения, анализа продуктов учебной деятельности, экспертной оценки и обобщения педагогического опыта.

В качестве базовой единицы анализа рассматривается учебный проект, содержащий исследовательскую задачу и предполагающий создание цифрового продукта. Методическая логика строится не вокруг заранее заданной инструкции, а вокруг последовательного усложнения самостоятельности учащегося. Учитель задает рамки и критерии, обеспечивает доступ к ресурсам и оказывает консультационную поддержку, однако выбор конкретного способа решения, проверка данных и обоснование результата выполняются учащимися.

Для оценки сформированности критического и алгоритмического мышления предлагается использовать совокупность процессуальных и результативных показателей. К процессуальным относятся самостоятельность постановки проблемы, качество поиска информации, логика планирования, обоснованность выбора инструмента, способность тестировать промежуточные решения и корректировать ошибки. К результативным показателям относятся работоспособность цифрового продукта, логическая целостность алгоритма, доказательность выводов, качество презентации и глубина рефлексии.

РЕЗУЛЬТАТЫ И ОБСУЖДЕНИЕ

На основе теоретического анализа разработана методическая модель развития критического и алгоритмического мышления на уроках информатики. Она включает шесть взаимосвязанных этапов, каждый из которых имеет собственную педагогическую функцию и направлен на формирование определенных мыслительных действий.

Таблица 1. Этапы проектно-исследовательского обучения на уроках информатики

Этап	Деятельность учителя	Деятельность учащегося	Критическое мышление	Алгоритмическое мышление
1. Мотивационно-проблемный	Создает проблемную ситуацию, задает контекст	Формулирует проблему и вопросы	Выявление противоречия, постановка вопросов	Определение входных данных и ожидаемого результата
2. Аналитико-поисковый	Направляет поиск, предлагает критерии надежности	Ищет, сравнивает и проверяет информацию	Оценка источников, сопоставление фактов	Выделение значимых данных и ограничений
3. Алгоритмический	Задает ориентиры для планирования	Декомпозирует задачу, строит алгоритм	Сравнивает варианты и обосновывает выбор	Последовательность, условия, циклы, декомпозиция
4. Практико-технологический	Консультирует по цифровым инструментам	Создает и тестирует цифровой продукт	Проверяет корректность промежуточных решений	Реализация, тестирование, отладка алгоритма
5. Презентационно-оценочный	Организует защиту и взаимное оценивание	Представляет решение, отвечает на вопросы	Аргументация и защита позиции	Объяснение логики алгоритма и его эффективности
6. Рефлексивный	Организует самооценку и анализ ошибок	Оценивает процесс и планирует улучшения	Критическая самооценка и выводы	Оптимизация алгоритма, выявление избыточных действий

На мотивационно-проблемном этапе важно отказаться от формулировки задания как последовательности готовых действий. Учителю целесообразно предлагать ситуацию, в которой имеется реальная информационная потребность. Например: «Как автоматически определить, какие книги школьной библиотеки пользуются наибольшим спросом?» или «Как создать цифровую систему, которая поможет учащемуся планировать учебную нагрузку?». Такая постановка требует от школьника сначала понять проблему, а уже затем выбрать инструмент.

Аналитико-поисковый этап направлен на развитие информационной критичности. Учащиеся определяют, какие данные им нужны, откуда их можно получить, насколько они надежны, каким образом проверить их актуальность и непротиворечивость. Для уроков информатики особенно важно вводить критерии цифровой надежности: авторство источника, дата публикации, наличие подтверждений, совпадение сведений в нескольких источниках, корректность формата данных.

Алгоритмический этап является ядром предлагаемой модели. Учащийся должен преобразовать общую проблему в последовательность подзадач, определить входные и выходные данные, предусмотреть условия и возможные исключения. До программной реализации полезно использовать псевдокод, блок-схему, таблицу решений или словесный алгоритм. Такая предварительная формализация позволяет отделить логику решения от конкретного программного средства.

Практико-технологический этап предполагает непосредственное создание цифрового продукта. При этом работа не должна сводиться к механическому повторению действий учителя. Необходимо предусматривать выбор: среды программирования, структуры данных, способа визуализации, формы интерфейса или набора функций. После создания прототипа проводится тестирование на нескольких наборах данных. Обнаруженные ошибки рассматриваются не как неудача, а как источник анализа и совершенствования алгоритма.

На презентационно-оценочном этапе учащийся должен объяснить не только то, что он создал, но и почему выбрал именно такой способ решения. Вопросы «Какие альтернативы рассматривались?», «Почему выбранный алгоритм эффективнее?», «Какие ограничения имеет решение?», «Какие данные могут привести к ошибке?» переводят защиту проекта из демонстрации продукта в процедуру аргументированного научного обсуждения.

Рефлексивный этап завершает цикл. Учащийся анализирует, какие действия оказались наиболее результативными, где возникли затруднения, какие решения были избыточными, как можно улучшить алгоритм и какие знания потребуются в следующем проекте. Благодаря этому опыт одной проектной работы превращается в основу для дальнейшего самостоятельного развития.

Таблица 2. Критерии оценки критического и алгоритмического мышления

Критерий	1 балл	2 балла	3 балла	4 балла
Постановка проблемы	Не понимает проблему без подсказки	Понимает после разъяснения	Самостоятельно формулирует проблему	Уточняет проблему, выделяет противоречия и ограничения
Анализ информации	Использует случайные сведения	Находит данные, но слабо оценивает надежность	Сравнивает источники и проверяет данные	Системно оценивает надежность, актуальность и достаточность данных
Построение алгоритма	Действия фрагментарны	Есть последовательность, но имеются логические пробелы	Алгоритм логичен и работоспособен	Алгоритм оптимизирован, учитывает альтернативы и исключения
Тестирование и отладка	Не выявляет ошибки	Исправляет после указания	Самостоятельно находит и исправляет ошибки	Разрабатывает систему тестов и объясняет причины ошибок
Аргументация	Не обосновывает решение	Приводит отдельные доводы	Логично объясняет выбор	Сравнивает альтернативы, аргументирует преимущества и ограничения
Рефлексия	Не анализирует процесс	Называет отдельные трудности	Делает выводы о результатах	Предлагает конкретные

				способы улучшения и дальнейшего развития

Суммарный результат по шести критериям может составлять от 6 до 24 баллов. Для педагогической диагностики предлагается следующая интерпретация: 6–11 баллов — репродуктивный уровень; 12–17 баллов — функциональный уровень; 18–21 балл — самостоятельный уровень; 22–24 балла — исследовательско-творческий уровень. Такая шкала не предназначена для замены традиционной отметки по предмету, а используется для мониторинга динамики мыслительных навыков.

ПРИМЕРЫ ПРОЕКТНО-ИССЛЕДОВАТЕЛЬСКИХ ЗАДАНИЙ

1. «Интеллектуальный планировщик учебного времени» (7–8 классы).

Учащиеся анализируют факторы, влияющие на распределение времени, определяют входные данные, создают алгоритм приоритизации заданий и реализуют прототип в электронной таблице, Scratch или Python. Исследовательский компонент заключается в сравнении нескольких правил приоритизации и выборе наиболее эффективного.

2. «Надежность информации в сети: автоматизированная памятка» (8–9 классы).

Учащиеся формируют критерии надежности интернет-источников, исследуют примеры достоверной и сомнительной информации, создают интерактивную форму или веб-страницу, помогающую пользователю оценивать источник по заданным признакам. Основное внимание уделяется критической оценке цифровой информации.

3. «Анализ школьных данных» (9–10 классы).

На обезличенном наборе данных учащиеся выполняют очистку, группировку и визуализацию, формулируют исследовательские вопросы, строят диаграммы и делают выводы. Важным условием является обсуждение того, какие выводы допустимы, а какие нельзя делать из имеющихся данных.

4. «Оптимальный маршрут внутри школьного здания» (10 класс).

Учащиеся представляют план здания в виде графа, определяют вершины и ребра, сравнивают возможные маршруты и разрабатывают алгоритм поиска подходящего пути. Проект развивает навыки моделирования, декомпозиции и оценки эффективности алгоритмов.

5. «Мини-система обнаружения ошибок в тексте или данных» (10–11 классы).

Учащиеся формулируют правила выявления типичных ошибок, создают алгоритм проверки, реализуют его программно и тестируют на специально подготовленных наборах данных. В ходе защиты анализируются ложные срабатывания, ограничения алгоритма и направления его улучшения.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

1. Проектная задача должна содержать открытый вопрос или проблему, допускающую несколько вариантов решения.

2. Перед практической реализацией следует требовать от учащихся краткий план исследования и предварительный алгоритм.

3. Поиск информации должен сопровождаться критериями проверки надежности источников.

4. Для каждого проекта рекомендуется предусматривать не менее двух этапов тестирования: промежуточный и итоговый.

5. В оценивание необходимо включать не только качество продукта, но и логику решения, самостоятельность, аргументацию и рефлексию.

6. Учителю следует постепенно уменьшать объем прямых инструкций по мере роста самостоятельности учащихся.

7. Групповая работа должна сочетаться с индивидуальной ответственностью: каждый участник объясняет собственный вклад и понимает общий алгоритм проекта.

8. Целесообразно формировать цифровое портфолио, в котором сохраняются постановка проблемы, версии алгоритма, результаты тестирования, итоговый продукт и рефлексивный комментарий.

ЗАКЛЮЧЕНИЕ

Проведенный теоретико-методический анализ показывает, что проектно-исследовательское обучение обладает значительным потенциалом для развития критического и алгоритмического мышления учащихся на уроках информатики. Его эффективность определяется тем, что школьник включается в полный цикл интеллектуальной деятельности: от осмысления проблемы и поиска информации до построения алгоритма, цифровой реализации, тестирования, защиты и рефлексии.

Предложенная шестиступенчатая модель позволяет целенаправленно распределить учебные действия и связать каждый этап проекта с конкретными мыслительными навыками. Наиболее существенным является переход от обучения по готовому образцу к управляемой самостоятельности, когда учитель сохраняет педагогическую поддержку, но не подменяет собой интеллектуальную деятельность учащегося.

Комплексное оценивание по критериям постановки проблемы, анализа информации, построения алгоритма, тестирования, аргументации и рефлексии дает возможность отслеживать не только конечный результат, но и качество мыслительного процесса. В перспективе данную методику целесообразно апробировать на различных возрастных группах, сопоставить результаты проектных и традиционных форм обучения, а также разработать цифровые инструменты мониторинга индивидуальной динамики учащихся.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Dewey J. Experience and Education. — New York: Macmillan, 1938.
2. Kilpatrick W. H. The Project Method // Teachers College Record. — 1918. — Vol. 19, No. 4. — P. 319–335.
3. Blumenfeld P. C., Soloway E., Marx R. W., Krajcik J. S., Guzdial M., Palincsar A. Motivating Project-Based Learning: Sustaining the Doing, Supporting the Learning // Educational Psychologist. — 1991. — Vol. 26, No. 3–4. — P. 369–398.
4. Thomas J. W. A Review of Research on Project-Based Learning. — San Rafael, CA: Autodesk Foundation, 2000.

5. Bell S. Project-Based Learning for the 21st Century: Skills for the Future // The Clearing House. — 2010. — Vol. 83, No. 2. — P. 39–43.
6. Wing J. M. Computational Thinking // Communications of the ACM. — 2006. — Vol. 49, No. 3. — P. 33–35.
7. Papert S. Mindstorms: Children, Computers, and Powerful Ideas. — New York: Basic Books, 1980.
8. Markham T. Project Based Learning Design and Coaching Guide. — San Rafael, CA: HeartIQ Press, 2012.
9. Государственная учебная программа по информатике и информационным технологиям для общеобразовательных школ Республики Узбекистан.
10. Очилова Д. А. Использование технологии проектного обучения на уроках информатики как средство развития самостоятельного мышления учащихся: авторская методическая разработка.