

Развитие регулятивных УУД в процессе обучения программированию

Основной задачей современной системы образования является формирование совокупности «универсальных учебных действий», которые обеспечивают возможность каждому ученику самостоятельно осуществлять деятельность учения, ставить учебные цели, искать и использовать необходимые средства и способы их достижения, уметь контролировать и оценивать учебную деятельность и ее результаты. Они создают условия развития личности и самореализации.

УУД направлены на достижение планируемых результатов, которые делятся на три группы:

1. Предметные УУД – основа изучения самого предмета (опыт получения, преобразования и применения предметных знаний).

2. Метапредметные УУД – центральная составляющая формирования умений у обучающихся работать с информацией (извлекать её, анализировать, воспринимать), отражающие межпредметные понятия.

3. Личностные УУД – эмоциональность и нравственность в изучении предмета, развитие толерантности, здорового образа жизни.

Группу метапредметных УУД составляют:

- Регулятивные УУД обеспечивают учащимся организацию их учебной деятельности.
- Познавательные УУД включают: общеучебные, логические, а также постановку и решение проблемы.
- Коммуникативные УУД обеспечивают социальную компетентность и учет позиции других людей, партнеров по общению или деятельности; умение слушать и вступать в диалог; участвовать в коллективном обсуждении проблем; интегрироваться в группу сверстников и строить продуктивное взаимодействие и сотрудничество со сверстниками и взрослыми.

Курс информатики имеет ярко выраженную направленность на развитие и совершенствование системы универсальных учебных действий. Нас интересует группа регулятивных УУД, которые обеспечивают возможность управления познавательной и учебной деятельностью посредством постановки целей, планирования, контроля, коррекции своих действий и оценки успешности усвоения. Умение ставить личные цели, понимать и осознавать смысл своей деятельности, при этом, соотнося его с требованиями внешнего мира, определяет в значительной степени успех личности вообще и успех в образовательной сфере в частности.

Так, например, обучающиеся 5 – 6 классов должны уметь составлять план в соответствии с поставленной задачей, в 7 – 8 классах – уметь составлять план деятельности (цель, прогнозирование, контроль), ну, а обучающиеся 9 – 11 классов уже должны уметь планировать и осуществлять деятельность, направленную на решение задач исследовательского характера.

Данные компетенции, а именно умение самостоятельно планировать пути достижения цели, в том числе альтернативные, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач, обеспечиваются алгоритмической линией, которая реализована в учебниках в разделе «Основы алгоритмизации и объектно-ориентированного программирования». Обучающиеся при со-

ставлении алгоритма и разработке программы сталкиваются с необходимостью регуляции своих действий.

Процесс написания алгоритма или программы можно разделить на этапы: постановка задачи, анализ задачи, разработка алгоритма, программирование, тестирование и отладка, анализ результатов решения задачи и сопровождение программы. Рассмотрим соотношение этапов решения задач по программированию с действиями учащихся:

Этап	Предмет действия	Цель действия	Система операций	Ориентировочная основа	Результат
Постановка задачи.	Текст задачи	Определение конечной цели программы.	Анализ текста задачи. Определение исходных данных, их ограничений. Определение выходных данных.	Анализ возможных решений, условий функционирования программы.	Четкая формулировка решаемой задачи.
Математическая формализация (анализ задачи).	Математическая модель задачи (необходимые формулы, расчеты, алгоритмы и др.)	Формализация условий задачи	Выделить существенные свойства объекта. Оптимизировать требования задачи. Подобрать необходимые формулы	Контроль адекватности применения выбранных методов и средств обучения	Создание математической модели
Разработка алгоритма, дающего решение задачи.	Блок-схема/ Псевдокод/ План решения	Разработка алгоритма	Определить последовательность шагов решения задачи, выполнить трассировку, частично формализовать алгоритм	Общие правила и рекомендации составления наглядной схемы алгоритма	Построение алгоритма в удобной для восприятия, наглядной форме
Составление программы. Программирование.	Язык программирования	Написание программы с соблюдением синтаксических и семантических правил	Записать общую структуру программы, перевести созданный на предыдущем этапе алгоритм на язык программирования	Синтаксические и семантические правила программирования	Программа на выбранном языке
Отладка и тестирование программы.	Среда программирования, интерпретатор	Проверка корректности программы	Проверить синтаксические ошибки, выполнить контрольный пример	Средства проверки корректности программы	Интерпретация программы исполнителем
Анализ полученных результатов.	Программа, исполнитель	Верификация полученных результатов	Проанализировать работу программы при разных исходных данных	Соотнесение кода и действий исполнителя	Получение учащимися новых знаний

Педагогу необходимо подобрать для каждого обучающегося задания, способствующие развитию у него регулятивных УУД, соотнося эти задания с его способностями. Помимо этого задания могут быть как индивидуальные, так и групповые.

Знакомство с алгоритмами и азами программирования можно начать со следующего типа заданий:

Задание «Сказка». Цель: осуществление контроля действий, описанных в сказках, написание алгоритма. Описание задания: ученикам предлагается выбрать сказку, о которой можно сказать, что в ней действие описывается по алгоритму с циклом? Если ученики не могут вспомнить сами сказку им предлагается следующий список сказок: Сказка о мертвой царевне и семи богатырях; Снегурочка; Курочка Ряба; Сказка о рыбаке и рыбке. Необходимо выбрать сказку и описать алгоритм в удобном виде.

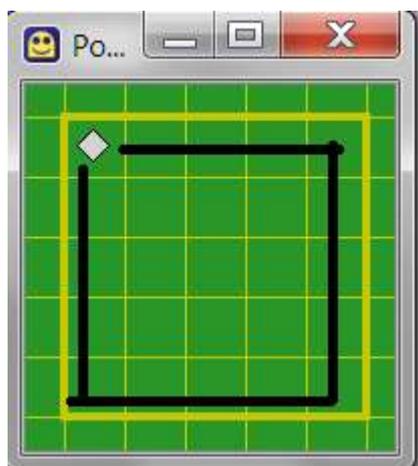
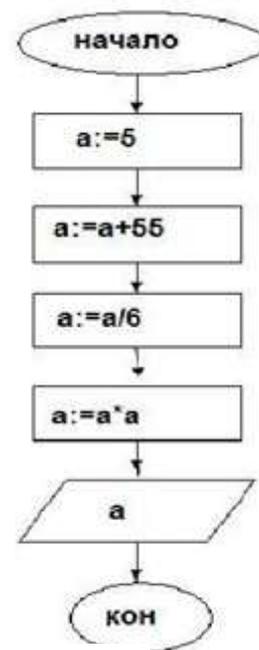
Сказку можно заменить стихами или другими произведениями, содержащими в себе алгоритм.

Далее задания усложняются, например, «**Выполнить алгоритм**». Цель: формирование умения составления и выполнения алгоритма. Формирование алгоритмического типа мышления. Описание задания: А) Какое значение получит переменная a после выполнения следующего алгоритма? Б) запишите, выражение, для которого был составлен алгоритм.

На следующем этапе освоения раздела предлагается тип заданий «**Найди ошибку**», по результатам которого обучающийся проверяет уровень усвоения темы и развивает умение анализировать работу программы.

Описание задания: Ученик торопился и при записи алгоритма для Робота допустил ошибку. Если ты не исправишь ошибку, Робот «разобьется».

Условие: робот должен пройти по периметру квадрата со стороной 5 клеток и вернуться в исходную клетку.



```
1  использовать Робот
2  алг Робот
3  нач
4  - если справа свободно то вправо
5  - все
6  - если снизу свободно то вниз
7  - все
8  - если слева свободно то влево
9  - все
10 - если сверху свободно то вверх
11 - все
12 кон
```

Ученик написал следующий алгоритм. Исправь его.

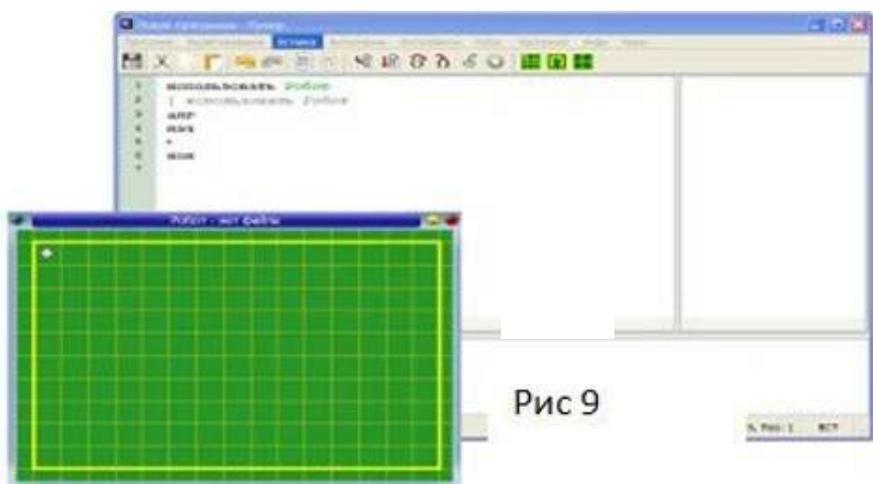
алго-

Результат исправления: В данном случае надо заменить связку «если ... свободно то ...», на «пока ... свободно то...».

После изучения линейных алгоритмов, вводятся алгоритмы с циклом и ветвлением. На данном этапе можно использовать задания типа «**Алгоритм с ветвлениями (циклом)**». Цель: формирование умения составления и выполнения алгоритма. Формирование алгоритмического типа мышления. Описание задания: найдите структуру «ветвление» в стихах известных поэтов и запишите ее, например

В.Маяковского «Что такое хорошо и что такое плохо?» или А.С.Пушкина «Сказка о царе Салтане, о сыне его славном и могучем богатыре князе Гвидоне Салтановиче и о прекрасной царевне Лебеди».

На следующем этапе изучения и подготовки обучающихся к освоению объектно-ориентированного программирования можно использовать задания для системы программирования КуМир: по рисунку написать программу и провести ее отладку. Развитие умения определять последовательность шагов решения задачи, частичная формализация алгоритма. Порядок выполнения работы в приложении Ку-



Мир:

1. Запустить приложение КуМир (окно приложения уже содержит шаблон алгоритма)
2. Меню Вставка – использовать Робот
3. Напечатать название алгоритма.
4. Провести анализ рисунка (определить порядок его выполнения)
5. Напечатать алгоритм (на-

пример):

алг рисунок

нач

вправо; закрасить; вправо; закрасить; вправо
вниз; закрасить; вниз; закрасить; вниз
влево; закрасить; влево; закрасить; влево
вверх; закрасить; вверх; закрасить.

кон

6. Провести отладку программы.

Далее в 9 – 11 классах все задания имеют уровневую составляющую, которая предопределяет степень развития как регулятивных, так и предметных УУД обучающихся:

- 1 уровень: Найдите синтаксическую ошибку в записи (в этом случае учителю стоит попросить сопоставить запись программы на алгоритмическом языке и на языке Pascal, ошибка может быть как грубая, так и незначительная, то есть не влияющая на выполнение программы).
- 2 уровень: Исправьте ошибку в программе в соответствии со схемой, таблицей или условием (учитель может предложить разные материалы, затем учащиеся исправят ошибки и изучат различия между получившимися программами.)
- 3 уровень: Измените условие задачи так, чтобы найти только ... или ...; измените задачу так, чтобы найти ... и ... (в данном случае учитель может сначала попросить изменить условия в АЯ, а затем написать на ее основе программу.)

В процессе изучения раздела у обучающихся регулятивные УУД развиваются постепенно, поэтому необходимо отслеживать их развитие, они могут это делать под руководством педагога, но лучше, если они самостоятельно будут себя оцени-

вать по предложенным критериям. При таком подходе у них развивается еще одно универсальное действие как самоконтроль, которое относится к регулятивным УУД.

Мониторинг развития регулятивных УУД			
№	Критерий	Эталонный балл	Балл учащегося
1	Нашел все ошибки	0 – 1	
2	Исправил ошибки верно и корректно	0 – 3	
3	Составил схемы и таблицы	0 – 2	
4	Написал программу, соответствующую представленным учителем материалом и языку программирования, рационально и верно	0 – 3	
5	Составил схемы оптимально	0 – 2	
6	Изменил условие задачи в соответствии с требованиями учителя	0 – 3	
7	Составил программу, которая соответствовала условиям задачи, была рациональна записана согласно выбранному языку программирования	0 – 3	

Задания, разработанные на основе типовых задач, и уровневые задания необходимо систематически использовать на уроке, что станет мощным фактором развития регулятивных УУД. А само развитие регулятивных УУД, так же, как и других метапредметных УУД, необходимо проводить на основе деятельностного подхода, основываясь на дифференцированном, исследовательском, компетентностном, развивающем и проблемном обучении.

Следовательно, учитель сначала оценивает интерес учащихся, затем дает установку на каждом уроке с помощью типовых заданий и листов оценивания, заинтересовывает через представление теории с помощью рассказа об успешных программистах, видео, картинок с блок-схемами, интернет-сервисов и разноуровневых заданий.

Потом обучающиеся, переходя в старшее звено (10 – 11 класс), способны сознательно организовать и регулировать свою деятельность: учебную, общественную и др., контролировать и корректировать ее, давать ей оценку, а также самостоятельно ставить цели, выбирать и создавать алгоритмы для решения учебных проблем, организовывать сотрудничество для их решения. Помимо этого, в зависимости от класса учитель так же дает возможность сильным обучающимся помогать более слабым.

Только при таком системном подходе можно добиться результата.

Источники:

1. Статья Гребнева Д.М. Обзор методических подходов к обучению программированию в школе // Научное обозрение. Педагогические науки. – 2016. – № 3. – С. 13-27; URL: <https://science-pedagogy.ru/ru/article/view?id=1495> (дата обращения: 13.04.2018).
2. Статья Брославская Т. Л. Формирование и развитие регулятивных УУД у обучающихся на уроках русского языка // Молодой ученый. — 2015. — №13. — С. 605-608. — URL <https://moluch.ru/archive/93/20554/> (дата обращения: 13.04.2018).
3. Статья Н. В. Трифонова, Материалы I Всероссийской очно-заочной практической конференции «Математика, физика, информатика: проблемы и перспективы современного образования» (Новокузнецк, февраль 2016).