Н. В. Трифонова

ФОРМИРОВАНИЕ РЕГУЛЯТИВНЫХ УНИВЕРСАЛЬНЫХ УЧЕБНЫХ ДЕЙСТВИЙ ПРИ ИЗУЧЕНИИ ОСНОВ ПРОГРАММИРОВАНИЯ В 9 КЛАССАХ

В современном обществе все больше нарастает поток информации, который необходимо обрабатывать человеку. Поэтому социальный заказ государства изменился, теперь нужно не только знать, уметь и обладать навыками, но и быстро и правильно реагировать в изменяющейся обстановке, а для этого «уметь учиться» особенно важно.

Чтобы выполнить социальный заказ общества, был разработан новый стандарт образования, в котором прописаны метапредметные результаты. Так, в разделе «Основы программирования» учитель должен уделить особое внимание регулятивным УУД, так как учащиеся при разработке программы сталкиваются с необходимостью регуляции своих действий при ее написании: постановка задачи, анализ задачи, разработка алгоритма, программирование, тестирование и отладка, анализ результатов решения задачи и сопровождение программы.

В ходе работы с учащимися было выявлено, что интерес и мотивацию в обучении программированию необходимо поддержать с помощью современных технологий как педагогических, так и информационных с учетом системно-деятельностного подхода. Так, вовлечение учащихся в деятельность позволяет учителю также реализовывать «план Трампа»: наиболее успешно работающие учащиеся начинают выполнять роль помощников учителя (в белль-ланкастерской системе взаимного обучения этих учеников – помощников учителя – называют мониторами).

Необходимо помнить, что при составлении заданий учитель отталкивается от уровня класса, заинтересованности учащихся и статуса предмета для данного класса (интегрированный, базовый и профильный).

Формирование регулятивных УУД, так же, как и других метапредметных УУД, необходимо проводить на основе деятельностного подхода, что предполагает активность обучающихся. Поэтому в ходе проектирования урока учитель должен использовать методы, основываясь на дифференцированном, исследовательском, компетентностном, развивающем и проблемном обучении.

Исследовательские и проблемные методы можно применять при анализе задач, прогнозировании, контроле и коррекции их решения, а при отборе задачи учитель должен руководствоваться личностными интересами учащихся и интеграции с математикой.

В начале изучения темы «Программирование» учителю целесообразно представить этапы создания программы с помощью «петли качества» Эдварда Деминга - «Plan-Do-Check-Act» - планирование-действие-проверка-корректировка).

Способ действий по данной концепции при первом знакомстве с этапами создания программы учащиеся лучше усвоят, затем постепенно, по мере изучения программирования ее этапы нужно будет сопоставить с этапами создания программы.

Пример задачи: Определить объём и площадь боковой поверхности цилиндра с заданными радиусом основания R и высотой H.

Данные	Результат
R= 5; H=4	V=314,16; S=125,6
R= 7;H=10	V=1539,38; S=439,6
R= 15; H=3	V=2120,58; S=282,6

```
Program Cylinder;

  Uses Crt; {Подключение библиотеки Crt}

  Var

    R, {радиус основания цилиндра}

    H, {высота цилиндра }

    V, {объем цилиндра }

    S: Real; {площадь боковой поверхности цилиндра}

BEGIN

  ClrScr; {Вызов из библиотеки Crt процедуры очистки экрана}

  Write ('Введите высоту цилиндра: ');  ReadLn(H);

  Write ('Введите радиус основания: '); ReadLn(R);

  V := Pi * R * R * H;

  S := 2 * Pi * R * H;  WriteLn;

   WriteLn ('Объем цилиндра = ', V: 5: 2); {Здесь 5 - общее количество позиций, занимаемых переменной V при выводе, а 2 - количество позиций в дробной части значения V}

  WriteLn ('Площадь боковой поверхности = ', S: 5: 2);

  ReadLn;

    END.
```

Примеры заданий:

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 площадь или объем; измените задачу так, чтобы найти объем и площадь боковой поверхности пирамиды (в данном случае учитель может сначала попросить изменить условия в АЯ, а затем написать на ее основе программу.)

Регулятивные УУД

Умение оценивать правильность выполнения учебной задачи

Умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата, определять способы действий в рамках предложенных условий, корректировать свои действия в соответствии с изменяющейся ситуацией.

Умение оценивать правильность выполнения учебной задачи

Умение самостоятельно планировать пути достижения целей.

Умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата, определять способы действий в раках предложенных условий и требований, корректировать свои действия в соответствии с изменяющимися ситуацией.

Умение оценивать правильность выполнения учебной задачи.

В ходе изучения раздела учитель должен иметь в виду необходимость постепенного формирования у учащихся всех регулятивных УУД. Целесообразно использовать критерии, чтобы учащиеся имели представления о своих ошибках и могли оценивать свою учебную деятельность. Их необходимо показать учащимся в начале урока.

Критерии оценивания

	Баллы
Учащиеся нашли все ошибки	0-1
Учащиеся нашли все ошибки	0-1
Исправили ошибки верно и корректно	0-3
Составили схемы и таблицы	0-2
Написали программу, соответствующую представленным учителем материалам и языку программирования, рационально верно	0-3
Изменили условия задачи в соответствии с требованиями учителя	0-3
Составили схемы оптимально верно	0-2
Составили программы, которые соответствовали условиям задачи, были рациональными, записаны согласно выбранному языку программирования	0-3

Изучение основам программирования на сегодняшний день стало доступнее. Поэтому на уроках так же стоит уделить внимание использованию Интернет-источников, например обучающее видео, интернет-сервисы (например blockly от google – удобный сервис, в котором программирование осуществляется блок-схемами с возможностью визуализации результата; программа Alice и др.). Так с одной стороны мы обеспечим понимание темы, а с другой - развитие интереса.

Только при таком системном подходе возможно добиться результата. Таким образом, учитель сначала оценивает интерес учащихся, затем дает установку на каждом уроке с помощью листов оценивания, заинтересовывает через представлении теории с помощью рассказа о успешных программистах, видео, картинок с блок-схемами(см. картинки: их преобразованием и созданием на уроке по частям в настоящие блок-схемы: исправляем ошибки, создаем свои), интернет-сервисов и разноуровневых заданий. В зависимости от класса учитель так же дает возможность сильным учащимся помогать более слабым.

ИНФОРМАЦИОННЫЕ ИСТОЧНИКИ

1. Извозчиков В.А. Информология, информатика и образование. Справочное пособие.- М.: Каро., 2003.- С. 32.

2. Кинилев В.Г. Контуры системы образования в XXI веке. Журнал «Информатика и образование». 2000. №5.

3. Колин К.К. Социальная информатика – новое направление научных исследований. /Сб. научн. трудов «Системы и средства информатики». – М.: Наука, 2005, вып.7.- С. 20-37.

4. Кузнецов А.А. Основы информатики, 8-9 –е классы. М: Дрофа, 2003.

5. Ракитов А.И. Информация, наука, технология в глобальных исторических измерениях.– М.: ИНИОН РАН, 2008.- С. 87.

6. Соколова И.В. Социальная информатика и социология: проблемы и перспективы взаимосвязи. М.: Владос, 2003.- С. 52.