**УДК 372.8**

**РОБОТОТЕХНИКА КАК СРЕДСТВО ФОРМИРОВАНИЯ АЛГОРИТМИЧЕСКОГО МЫШЛЕНИЯ**

**Даерова Евгення Викторовна**

**учитель математики,**

**МБОУ "Ивановская СОШ" Основное общее образование**

Алгоритмическая культура как часть информационной культуры предполагает формирование и развитие у учащихся представлений и умений, связанных с пониманием сущности алгоритма, его свойств и методов записи, владением приёмами решения задач на алгоритмическом языке. Создание алгоритма – специальный мыслительный навык, которым люди изначально не обладают. Алгоритмическое мышление рассматривается как представление последовательности действий. Данный стиль мышления наряду с образным и логическим мышлением определяет интеллектуальную силу человека и его творческий потенциал. Умение планировать, давать полное описание своих действий способствует формированию не только навыков разработки алгоритмов решения самых разных классов задач, но и мыслительных навыков, полезных в разных предметных и научных областях.

В приказе Минобрнауки России от 17.12.2010 N 1897 (ред. от 11.12.2020) "Об утверждении федерального государственного образовательного стандарта основного общего образования" устанавливаются следующие предметные результаты обучения в области математики и информатики, которые устанавливают актуальность развития алгоритмического мышления [1]:

1) формирование информационной и алгоритмической культуры; формирование представления о компьютере как универсальном устройстве обработки информации; развитие основных навыков и умений использования компьютерных устройств;

2) формирование представления об основных изучаемых понятиях: информация, алгоритм, модель - и их свойствах;

3) развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составить и записать алгоритм для конкретного исполнителя; формирование знаний об алгоритмических конструкциях, логических значениях и операциях; знакомство с одним из языков программирования и основными алгоритмическими структурами - линейной, условной и циклической;

Одним из перспективных направлений для развития алгоритмического мышления является внедрение в образовательный процесс робототехнического направления. Под термином образовательная робототехника подразумевается современное междисциплинарное направление обучения школьников, интегрирующее знания о физике, технологии, математике и ИКТ, позволяющее вовлечь в процесс инновационного научно-технического творчества учащихся разного возраста. Она направлена на развитие у молодежи навыков практического решения актуальных инженерно-технических задач и работы с техникой [2, с. 17]. С педагогической точки зрения, использование подобного направления имеет ряд важных достоинств.

Во-первых, это стимулирование мотивации учащихся к получению знаний. При работе с конструктором учащийся видит плоды своей работы и имеет возможность применить полученные знания на практике. Кроме того, работа по созданию робота предполагает активную творческую деятельность ребёнка. Это реализуется через решение нестандартных для учащегося задач и большое количество вариантов решения.

В-вторых, это формирование навыков программирования, развитие логического и алгоритмического мышления.

В условиях информатизации образования остро встаёт необходимость поиска новых подходов к развитию алгоритмических умений школьников. Старый подход к обучению школьников программированию при помощи только языков программирования и в редких случаях использования компьютерных исполнителей уже не отвечает реалиям сегодняшнего дня. Современное образование требует более активного внедрения робототехники в курс школьной программы. Всё вышесказанное свидетельствует о том, что использование робототехники в процессе обучения информатике позволяет перейти к изучению основ алгоритмического мышления, а также преодолеть недостатки традиционного подхода в обучении программированию и вывести его на новый уровень.

Курс изучения образовательной робототехники можно разделить на два блока: теоретический и практический. Их взаимосвязь должна быть неразрывной, тогда в ходе обучения можно формировать у учащихся способности к моделированию, конструированию и программированию более наглядно. Посредством наглядности имеется возможность «включить» ученика в активную деятельность. Организовать обучающее пространство таким образом, чтобы ребенок самостоятельно познавал какие-то процессы, явления.

Выделяются два пути проведения занятий с детьми [3, с. 56]:

1. апробирование имеющихся моторов и датчиков, понимание основных идей, освоение программного обеспечения, а после сборка робота и анализ его возможностей;
2. сборка робота, создание программы и последующее тестирование, проведение экспериментов, обращение за справками по работе моторов и датчиков, выполнение дополнительных заданий.

Рассмотрим развитие алгоритмического мышления на примере одной из задач – «Кегельринг» [4]. Наступает важный этап моделирования робота - перед нами стоят две взаимосвязанных задачи:

1. Разработка ключевого алгоритма поведения нашего робота;
2. Разработка механической конструкции робота, позволяющей реализовать требуемое поведение.

Рассмотрим следующую поведенческую модель:

1. Робот, находясь в центре поля, начинает вращаться по часовой стрелке, пока не заметит кеглю
2. Двигаясь в направлении кегли, робот выталкивает её за пределы окружности.
3. Заметив черную границу поля, робот возвращается назад, в место старта.

Следовательно, наш робот должен:

1. уметь вращаться на месте вокруг своей оси;
2. уметь двигаться прямолинейно;
3. уметь обнаруживать предмет, удаленный на некоторое расстояние;
4. уметь обнаруживать границу поля.

Данные требования диктуют нам условия конструкции робота:

1. для реализации первых двух условий подвижную платформу, использующую два больших мотора и вращающуюся опору (робот-тележка);
2. для обнаружения кегли воспользуемся одним из датчиков: инфракрасным или ультразвуковым;
3. границу поля нам поможет обнаружить датчик цвета.

После сборки робота приступим к созданию программы.

Подробно пропишем последовательность действий робота для обнаружения одной кегли на поле:

1. вращаться вокруг своей оси по часовой стрелке, пока впереди расположенный датчик не обнаружит кеглю;
2. остановиться напротив кегли;
3. двигаться вперед, пока датчик цвета не обнаружит черную границу поля;
4. остановиться;
5. двигаться назад в центр поля.

Приступим к реализации и отладке – научим нашего робота обнаруживать кеглю и останавливаться напротив неё. Сначала нам необходимо выбрать пороговое значение для обнаружения кегли напротив нашего робота. Подключим робота к среде программирования, затем установим его точно в центр поля, поставим напротив робота кеглю.

На "Странице аппаратных средств", находящейся в правом нижнем углу среды программирования, выберем вкладку "Представление порта» и снимем показание датчика, определяющего расстояние до кегли, установив соответствующий режим отображения показаний.

Теперь мы можем запрограммировать нахождение роботом кегли. Рассмотрим алгоритм действий, если у нас установлен ультразвуковой датчик:

1. Для того, чтобы заставить робота вращаться вокруг своей оси, воспользуемся программным блоком "Независимое управление моторами" "Зеленой палитры", Режим работы блока установим "Включить", значение мощности для одного порта установим равным 30, значение мощности для другого порта установим равным -30.
2. Для поиска кегли используем программный блок "Ожидание" "Оранжевой палитры" в режиме "Ультразвуковой датчик - Сравнение - Расстояние в сантиметрах".
3. После того, как робот окажется напротив кегли, используя программный блок "Независимое управление моторами" "Зеленой палитры" выключим моторы.

Также рассмотрим последовательность команд, если присутствует инфракрасный датчик:

1. Для того, чтобы заставить робота вращаться вокруг своей оси, воспользуемся программным блоком "Независимое управление моторами" "Зеленой палитры", Режим работы блока установим "Включить", значение мощности для одного порта установим равным 30, значение мощности для другого порта установим равным -30.
2. Для поиска кегли воспользуемся программным блоком "Ожидание" "Оранжевой палитры" в режиме "Инфракрасный датчик - Сравнение - Приближение".
3. После того, как робот окажется напротив кегли, используя программный блок "Независимое управление моторами" "Зеленой палитры" выключим моторы.

Загрузим получившуюся программу в робота и запустим её на выполнение. Переходим к реализации п. 3, 4 нашей последовательности действий.

Установим робота таким образом, чтобы датчик цвета находился точно над черной границей поля и измерим его значение в режиме "Яркость отраженного света".

Добавим к нашей программе обнаружения кегли следующие программные блоки:

1. Для того, чтобы заставить робота двигаться прямолинейно, воспользуемся программным блоком "Рулевое управление" "Зеленой палитры". Режим работы блока установим "Включить", параметр "Рулевое управление " = 0, параметр "Мощность" = 50.
2. Для поиска датчиком цвета черной границы воспользуемся программным блоком "Ожидание" "Оранжевой палитры" в режиме "Датчик цвета - Сравнение - Яркость отраженного сигнала", параметр "Тип сравнения" = 4, параметр "Пороговое значение" = 10.
3. После того, как робот пересечет черную линию, используя программный блок "Рулевое управление "Зеленой палитры" выключим моторы.

Осталось только научить его возвращаться в центр круга. "Страница аппаратных средств" среды программирования позволяет нам наблюдать не только текущие показания датчиков, но и накопленные показания датчика вращения мотора. Нажав, на значок мотора, мы можем установить предпочтительный нам вывод информации о вращении мотора в "Оборотах" или "Градусах". Программный блок "Вращение мотора" "Желтой палитры" позволяет получать и обрабатывать это значение в программе. Режим "Сброс" программного блока "Вращение мотора" устанавливает нулевое значение датчика и отсчет оборотов начинается сначала.

Воспользуемся этой возможностью: если мы сбросим показание одного из датчиков моторов в =0, то после остановки робота над черной линией, можно будет получить значение датчика в "Градусах" или "Оборотах" и, подав его на вход соответствующего параметра блока "Рулевое управление", заставить робота проехать точно такое же расстояние. А для того, чтобы робот двигался назад, необходимо изменить значение параметра "Мощность" на отрицательное значение.

Внесем необходимые изменения в нашу программу:

1. Перед началом движения вперед сбросим в 0 показания датчика вращения мотора, подключенного к одному из портов.
2. После остановки на черной границе поля считаем расстояние, пройденное мотором в градусах.
3. Полученное значение подадим в параметр "Градусы" программного блока "Рулевое управление", значение параметра "Мощность" = -50.

Программа готова.

Подобные задачи ставятся перед учащимися для развития алгоритмического мышления. На этапе анализа задачи ученики должны выделить как особенности конструкции, так и специальные программные блоки, применимые к конкретному решению задачи. Изменение условий задачи, её усложнение приводит к развитию таких мыслительных операций, как синтез и абстрагирование. Как следствие, у учащихся формируется алгоритмический стиль мышления, позволяющий решать другие виды задач.

Список источников

1. Приказ Минобрнауки России от 17.12.2010 N 1897 (ред. от 11.12.2020) "Об утверждении федерального государственного образовательного стандарта основного общего образования" (Зарегистрировано в Минюсте России 01.02.2011 N 19644) [Электронный ресурс]. — Режим доступа: http:// www.consultant.ru

2. Ваграменко, Я. А. Применение программируемых устройств с робототехническими функциями в учебном процессе // Я. А. Ваграменко, О. А. Шестопалова, Г. Ю. Яламов // Педагогическая информатика. — 2015. — № 2. — С. 16-28.

3. Еремёнок, А. П. Робототехника в общем и дополнительном образовании Псковской области как составляющая начального политехнического обучения / А. П. Еремёнок, Н. В. Яникова // Информатика и образование. — 2013. — № 9. — С. 55-58.

4. Помощь начинающим робототехникам [Электронный ресурс]. — Режим доступа: http://robot-help.ru