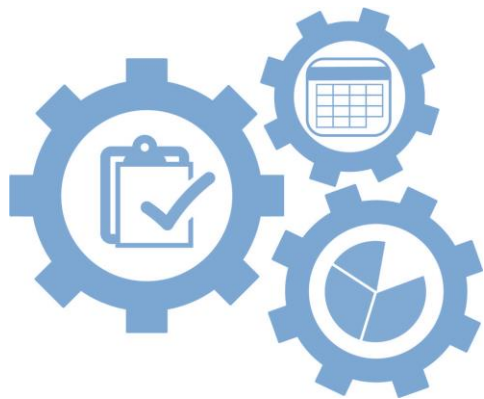


Программируем без компьютера: Введение в основы алгоритмики и робототехники с использованием образовательных роботов (Bee-Bot, Cubetto)

Тема 6. Лекция



Введение

Сегодня мы поговорим о теме, которая у многих ассоциируется со сложными кодами и экранами мониторов — о программировании. Но я хочу сразу вас успокоить: мы не будем писать код. Мы будем играть, творить и учить детей думать. Тема нашей встречи — **«Программируем без компьютера»**.

Мы живем в мире, где умение логически мыслить, планировать свои действия и находить ошибки становится важнее, чем простое запоминание фактов. И основы этого мышления, которое называют **алгоритмическим**, можно и нужно закладывать уже в дошкольном возрасте. Сегодня мы разберемся, что такое «алгоритм» на самом деле, зачем он нужен нашим воспитанникам, и познакомимся с двумя замечательными помощниками в этом деле — роботами-пчелкой Bee-Bot и деревянным роботом Cubetto.

Что такое алгоритм?

Понятие алгоритма: Это просто, как рецепт пирога!

Забудьте о сложных терминах. **Алгоритм** — это просто пошаговая инструкция, последовательность действий для достижения какой-то цели. Мы с вами сталкиваемся с алгоритмами каждый день:

Рецепт борща — это алгоритм. Если перепутать шаги (сначала посолить воду, а потом положить мясо), результат будет другим.

Инструкция по сборке мебели — это алгоритм.

Наш утренний ритуал (проснуться → умыться → одеться → позавтракать) — это тоже алгоритм.

Программа — это алгоритм, записанный на языке, понятном исполнителю. Для компьютера — это код, а для наших роботов — это последовательность нажатых кнопок или выложенных блоков.

Ключевые свойства алгоритма (объясняем на пальцах)

Дискретность (Пошаговость): Алгоритм состоит из отдельных, простых шагов. (*«Сначала сделай шаг вперед, ПОТОМ поверни направо»*).

Детерминированность (Определенность): При одних и тех же исходных данных результат всегда будет одинаковым. (Если мы нажмем «вперед-вперед-вправо», пчелка всегда приедет в одно и то же место).

Понятность: Все команды в алгоритме должны быть понятны исполнителю. (*Робот-пчелка не поймет команду «лети», она понимает только «шаг вперед»*).

Результативность (Конечность): Алгоритм всегда должен заканчиваться и приводить к какому-то результату. (*Пчелка должна дойти до цветка, а не ездить бесконечно*).

Зачем учить алгоритмике дошкольников?

Мы не растим программистов с пеленок. Мы развиваем **универсальные навыки (soft skills)**, которые пригодятся в любой сфере жизни:

Планирование: Прежде чем нажать кнопки, ребенок должен мысленно (или пальчиком на поле) проложить маршрут.

Логическое мышление: Ребенок учится выстраивать причинно-следственные связи.

Пространственное ориентирование: Понятия «вперед», «назад», «влево», «вправо» усваиваются не в теории, а на практике.

Навык «отладки» (поиска ошибок): Если пчелка уехала не туда — это не трагедия, а задача! «Где мы ошиблись? Какую команду нужно исправить?». Это учит анализировать свои действия и не бояться ошибок.

Коммуникация и работа в команде: Дети учатся договариваться, какой маршрут выбрать, и вместе составляют программу.

Знакомство с роботами: наши исполнители Bee-Bot («Умная Пчелка»)

Bee-Bot («Умная Пчелка»)

Внешний вид и философия: Яркая, дружелюбная пчелка. Идеальна для первого знакомства. Ее девиз: «Сначала думай, потом программируй». Программа вводится в память робота, а потом он ее выполняет.

Интерфейс (кнопки на спинке):

- ▶ ↑ (Вперед): Двигаться вперед на один шаг (стандартный шаг — 15 см).
- ▶ ↓ (Назад): Двигаться назад на один шаг (15 см).
- ▶ ← (Поворот влево): Повернуться на месте влево на 90 градусов.
- ▶ → (Поворот вправо): Повернуться на месте вправо на 90 градусов.
- ▶ || (Пауза): Сделать остановку на 1 секунду.
- ▶ GO (Зеленая): Запустить выполнение программы.
- ▶ X (Синяя, Сброс): Очистить память пчелки от предыдущей программы. **Это самая важная кнопка!** Всегда нажимаем ее перед составлением нового алгоритма.



Знакомство с роботами: наши исполнители Bee-Bot («Умная Пчелка»)

Принцип работы:

- ▶ Ставим пчелку на стартовую клетку.
- ▶ Нажимаем «X», чтобы очистить память.
- ▶ Продумываем маршрут и последовательно нажимаем кнопки команд (например, ↑, ↑, →, ↑).
- ▶ Нажимаем «GO».
- ▶ Наблюдаем, как пчелка выполняет программу. Если она ошиблась — ищем ошибку и начинаем заново.

Знакомство с роботами: наши исполнители Cubetto (Кубетто)

Внешний вид и философия: Тактильный, деревянный робот без экрана и кнопок. Состоит из двух частей: робота **Cubetto** и **Панели управления**. Его девиз: «Программирование, к которому можно прикоснуться».

Интерфейс (Панель и блоки):

Панель управления: Деревянная доска с углублениями, куда вставляются блоки.

Блоки команд (разного цвета):

- ▶ Зеленый: Шаг вперед.
- ▶ Красный: Поворот вправо на 90 градусов.
- ▶ Желтый: Поворот влево на 90 градусов.

Линия «Функция» (отдельная область на панели):

- ▶ Синий блок (Функция): Вставив его в основную программу, мы говорим роботу: «А теперь выполни то, что лежит в синей рамочке». Это простейшее введение в понятие подпрограммы. Например, если танец — это «вперед-поворот-вперед-поворот», мы можем выложить эту комбинацию в линию функции и вызывать ее одним синим блоком.



Знакомство с роботами: наши исполнители Cubetto (Кубетто)

Принцип работы

Ставим Cubetto на старт.

- ▶ Продумываем маршрут.
- ▶ Физически выкладываем цветные блоки на панель управления. Ребенок видит всю программу целиком.
- ▶ Нажимаем большую синюю кнопку «Старт» на панели.
- ▶ Наблюдаем. Если ошибка — меняем блоки местами, убираем лишние или добавляем нужные

Роль педагога: от наблюдателя к навигатору

Создание игровых полей и проблемных ситуаций.

Стандартные коврики — это хорошо, но творчество — лучше!

DIY-поля: Рисуем на большом листе ватмана город с домами, зоопарк, сказочный лес. Разметка поля — клетки 15x15 см для Bee-Bot или 20x20 для Cubetto.

Проблемные ситуации (сюжет): Не просто «доведи пчелку из точки А в точку Б».

«Пчелке нужно собрать нектар со всех красных цветов, не наступая на синие лужи».

«Cubetto — это пожарная машина. Нужно доехать до горящего дома самым коротким путем!»

«Давайте составим для пчелки программу танца: два шага вперед, поворот, два шага назад, поворот».

Какие вопросы педагог может задать ребенку?

Правильный вопрос важнее правильного ответа. Мы используем вопросы, чтобы направлять мышление ребенка:

На этапе планирования:

- ▶ *«Куда пчелке нужно поехать в первую очередь?»*
- ▶ *«Сколько шагов нужно сделать до поворота? Давай посчитаем вместе».*
- ▶ *«В какую сторону нужно будет повернуться — в сторону окна или в сторону двери?» (Привязка к реальным ориентирам).*

На этапе выполнения:

- ▶ *«Давай проверим, правильно ли едет пчелка? Считай ее шаги».*

На этапе "отладки" (если ошибка):

- ▶ *«Ой, пчелка уехала не туда. На каком шаге она ошиблась?»*
- ▶ *«Какую команду мы дали ей неправильно? Поворот или шаг вперед?»*
- ▶ *«Как ты думаешь, что нужно изменить в нашей программе?»*

Заключение

Коллеги, сегодня мы увидели, что алгоритмика — это не страшно, а увлекательно. Роботы Bee-Bot и Cubetto — это не просто дорогие игрушки. Это уникальные инструменты, которые позволяют сделать абстрактные понятия, такие как «план», «команда», «последовательность», видимыми, осязаемыми и понятными для ребенка.

Наша главная цель — не в том, чтобы ребенок безошибочно составлял программы, а в том, чтобы он научился думать, пробовать, ошибаться и исправлять свои ошибки с улыбкой. Мы учим его не кодированию, а образу мышления, который станет фундаментом его будущих успехов.