

«Создание игры «Поймай мяч!» на языке программирования Python»
Тихоновой Вилены Семеновны, ученицы 11 класса
МБОУ «Качикатская средняя общеобразовательная школа им.С.П.Барашкова»
Хангаласского района Республики Саха(Якутия).

В прошлом году я работала над созданием игры «Лыях» с помощью программы Scratch. В этом году решила изучить новую программу «Python».

Python – высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.

Актуальность – мы живем в век информатизации общества. Информационные технологии проникают в нашу жизнь с разных сторон. Одно из самых удивительных и увлекательных занятий настоящего времени - программирование. Изучение Python может помочь во всём, что угодно, - от веб-сайтов и игровых программ до управления роботами и космическими кораблями. Изучив программу, захотелось использовать Python для создания игры.

Итак, **цель** нашей работы – создание игры «Поймай мяч!» в среде программирования Python.

Для достижения цели мы поставили следующие **задачи**:

- изучить виды алгоритмов;
- изучить язык программирования Python;
- создать игру с помощью программы Python.

Язык программирования Python сочетает в себе программирование, ориентированный на повышение производительности разработчика и читаемости кода. Она оказалась очень увлекательной. Изучив программу, захотелось использовать Python для создания игры.

Описание игры.

«Поймай мяч!» - это компьютерная игра, в котором играет 2 игрока. В игре имеется: игровое поле, разделительное поле, мяч, 2 доски и счетчик очков. Правило данной игры совсем несложное, поэтому его без труда могут освоить даже младшие школьники.

В игре принимают участие 2 человека. В начале игры мяч сам начинает движение случайно с центра, так что надо быть всегда готовым.

1. Задача каждого участника – создать на поле ситуацию такую, чтобы мяч попал на сторону противника, но тот не сможет его отбить.
2. Победитель определяется по количеству очков, у кого больше отбитых мячей.

Ход работы:

1. Для начала создаём логин и пароль, чтобы войти в игру. Затем, импортируем библиотеки. Создаём окно, с размером 500x500 пикселей.
 - 1.1 Создаём функцию “Регистрация” с помощью текста “label” и пишем “Для входа в игру”. Создаём функцию “Вход в игру”, добавляем логин и пароль.
2. Затем, создаём поле с помощью “tkinter” с оконным интерфейсом. Добавляем глобальные переменные: “WIDTH-500 и HEIGHT-500” – это высота и ширина игрового поля.
 3. Задаём левую и правую доску «PAD_W=10 и PAD_H=100», создаем само окно, в котором будет сама игра, создаем имя игры и прорисовываем ровно в центре мячик «BALL_RADIUS=40».

4. Создаем модуль, в которой будут объединены классы. Делается это при помощи «Canvas». В «Canvas» создаются объекты-холсты, на которых можно "рисовать", размещать различные фигуры и объекты.

5. Создаём линию, в котором будет касаться мяч, за которую невозможно будет заступить. Создаём линию, которая делит поле на две части с помощью «с.create line».

6. Теперь приступаем к созданию команд для игры, т.е. движений мяча и досок.

6.1. Создаем движение мяча. Для этого добавляем переменную «скорость мяча, горизонтально и вертикально», затем создаем скорость мяча, горизонтальная скорость «X» = 20, по вертикали «Y»= 0 пиксель.

6.2. Делаем функцию движение досок. Мы называем «def_move_pads():». Создаем словарь, чтобы левая доска двигалась со скоростью левой доски, а правая со скоростью правой доски. Создаём переменную.

6.3. Создаём фокус на канвас. Для этого пишем функцию обработки на нажатии клавиш. Называем эту функцию, как «def moveent_handler(управление)» в параметре обязательно передаём событие «event».

6.4. Делаем отскок мяча. Создаём скорость мяча, с каждым ударом изначально будет увеличиваться на 1.00пиксель. Максимальная скорость мяча = 30пикселей, начальная скорость мяча по вертикали = 20, а по горизонтали тоже будет равен 20 пикселей.

7. Создаём очки. Создаём переменную «Очки для каждого игрока», «player_1_score = 0, player_2_score = 0» и «Текст очков», если 1 игрок пропустил мяч, то мы добавляем +1 очко 2 игроку, если 2 игрок пропустил мяч, то добавляем +1 очко 1 игроку.

8. Так же создаём «Респаун» для того, чтобы мяч после упущенных игроками, снова появился на центральной линии поля.

В результате у нас получилась интерактивная игра «Поймай мяч!».

Заключение.

Создание алгоритма анимации с помощью программы Python, развивает логическое мышление, творческую способность учащихся. В школе мы изучаем основы алгоритмизации и программирования, компьютерное моделирование, графику. В ходе работы познакомилась с очень интересной программой Python, которая сочетает в себе все эти темы. Особенностью этой среды является то, что в ней можно создавать мультики, анимации, приложения и даже простейшие игры. Для создания программных проектов Python имеет все необходимые средства, всё можно сделать здесь - в самой программе, без установки дополнительных программ.

Буду дальше изучать эту программу, так как она очень интересная, полезная и нужная. В дальнейшем планирую улучшить данную игру. А созданная игра очень понравилась моим друзьям и одноклассникам.