

<b>Школа:</b>		
<b>Дата:</b>	<b>ФИО учителя:</b>	
<b>Класс:</b>	<b>Участвовали:</b>	<b>Не участвовали:</b>
<b>Тема урока:</b> Цикл While, For		
<b>Цели обучения, которые достигаются на данном уроке</b>	1) Познакомить учащихся с циклом с предусловием и научить применять его при составлении программ на языке программирования Python. 2) Развитие мыслительной деятельности, речи, алгоритмического стиля мышления.	
<b>Цели урока</b>	Понять конструкцию «цикл WHILE» или «цикл с предусловием» Применять конструкцию WHILE для вывода фразы определенное количество раз Реализовать игры «Угадай число» и «Купи слона»	
<b>Критерии оценивания</b>	- Знают назначение и способ записи цикла WHILE - Умеют применять цикл WHILE для решения поставленной задачи	
<b>Воспитание ценностей</b>	Воспитание эмоционально-положительной направленности на практическую деятельность, интереса к информатике, личной ответственности за результаты своей работы.	
<b>Предварительные знания</b>	Типы данных, определение переменной, логические выражения, условный оператор if языка программирования Python	
<b>Межпредметные связи</b>	математика	
<b>Запланированные этапы урока</b>	<b>Запланированная деятельность на уроке</b>	<b>Ресурсы</b>
<b>Начало урока</b> <b>_7_ мин</b>	Приветствие учащихся. Психологический настрой на урок. Проверка и разбор домашнего задания Заданное на прошлом уроке домашнее задание: Написать программы: 1) Дано целое число. Если оно является положительным, то умножить его на 3; в противном случае вычесть из него 100. Вывести полученное число. 2) Определить, является ли число <b>a</b> делителем числа <b>b</b> . 3) Определить возможность существования треугольника по сторонам. (Треугольник существует только тогда, когда сумма любых двух его сторон больше третьей).	

Середина урока  
\_24\_ мин

1. Сегодня мы изучим конструкцию «цикл WHILE» или «цикл с предусловием» и попробуем написать первые игры.

WHILE – «пока» в переводе с английского

Посмотрим на пример цикла WHILE:

```
n=0
```

```
while n<3:
```

```
    n=n+1
```

Цикл WHILE означает следующее:

Пока <выполняется условие>: делать какие-то действия.

«Пока n меньше 3, прибавлять к n единицу»

Цикл повторяется, пока условие истинно, если же нет, цикл заканчивается.

Скажите, чему будет равно n после завершения цикла? (Ответ: 3)

Для того, чтобы увидеть, что происходит в теле цикла, создадим и запустим следующую программу:

```
n=0
```

```
while n<5:
```

```
    n=n+1
```

```
    print(n)
```

Программа выведет:

1

2

3

4

5

На первом шаге цикла n=1, на втором n=2 и так далее. Когда n станет равным пяти и программа выведет число 5, снова будет проверяться условие. Но условие не будет выполняться, так как  $5 < 5$  - не верно. И произойдёт выход из цикла.

Презентация

Пример:

```
n=0
while n<3:
    n=n+1
```

Чему будет равно n  
после завершения  
цикла?

4 пробела!

Запустить программу  
на компьютере:

```
n=0
while n<5:
    n=n+1
    print(n)
```

## 2. Запись в тетрадь:

while – «пока» в переводе с английского

Общая форма записи:

```
while <условие>:
```

```
    <действие 1>
```

```
    <действие 2>
```

и т.д.

Пример:

```
n=0
while n<5:
    n=n+1
    print(n)
```

### Запись в тетрадь!

WHILE – «пока» в переводе с английского

Общая форма записи:

```
while <условие>:
```

```
    <действие 1>
```

```
    <действие 2>
```

и т.д.

Пример:

```
n=0
while n<5:
    n=n+1
    print(n)
```

Бесконечный цикл.

```
while True:
    print("У попа была собака, он её
```

```
любил.")
```

```
    print("Она съела кусок мяса, он её  
убил,")
```

```
    print("В землю закопал и на камне  
написал:")
```

Пишем и запускаем эту программу на компьютере, чтобы увидеть бесконечный цикл в действии.

Запись в тетрадь:

```
Бесконечный цикл:  
while True:  
    <действия>
```

### Бесконечный цикл

Пример:

```
while True:  
    print("У попа была собака, он её любил.")  
    print("Она съела кусок мяса, он её убил,")  
    print("В землю закопал и на камне написал:")
```

Запустить эту программу на компьютере.  
(Чтобы приостановить выполнение программы,  
можно щёлкнуть по тексту правой кнопкой мыши).

Запись в тетрадь!

```
Бесконечный цикл:  
while True:  
    <действия>
```

Досрочный выход из цикла

```
break
```

Пример:

```
import random  
while True:  
    a=random.randint(1,10)  
    print(a)  
    if a==7:  
        break
```

Пишем и запускаем эту программу на компьютере, чтобы увидеть оператор break в действии.

Что происходит в программе?

## Досрочный выход из цикла

`break`

Пример:

```
import random
while True:
    a=random.randint(1,10)
    print(a)
    if a==7:
        break
```

Запустить программу на компьютере.

3. Учащиеся пишут программы на компьютерах под руководством учителя.

Задачи:

- 1) С помощью цикла `while` вывести любую фразу 7 раз.
- 2) Написать программу, которая получает два целых числа  $A$  и  $B$  ( $0 < A < B$ ) и выводит все натуральные числа в интервале от  $A$  до  $B$ .
- 3) С помощью бесконечного цикла реализовать игру «Купи слона». Программа должна выводить фразу «Купи слона!», получать ответ пользователя и выводить: «Все говорят “фраза, введённая пользователем”. А ты купи слона!» Затем программа получает новый ответ от пользователя и так до бесконечности.
- 4) С помощью бесконечного цикла и оператора `break` реализовать игру «Угадай число». Программа генерирует случайное число в диапазоне от 1 до 7 и говорит пользователю: «Угадай число от 1 до 7!». В бесконечном цикле программа считывает ответы пользователя и подсказывает ему «больше!» или «меньше!», а в случае правильного ответа цикл завершается, и выводятся сердечные поздравления с победой.

5) Доработать программу «Угадай число» так, чтобы она не заканчивалась, то есть, чтобы после угадывания одного числа, начиналось угадывание следующего. Увеличить диапазон с 7 до 15 и ввести ограничение на количество попыток (например, 3 попытки). Если пользователь не справился за 3 попытки, вывести «GAME OVER».



#### Задачи:

- 1) С помощью цикла `while` вывести любую фразу 7 раз.
- 2) Написать программу, которая получает два целых числа  $A$  и  $B$  ( $0 < A < B$ ) и выводит все натуральные числа в интервале от  $A$  до  $B$ .
- 3) С помощью бесконечного цикла реализовать игру «Купи слона». Программа должна выводить фразу «Купи слона!», получать ответ пользователя и выводить: «Все говорят “фраза, введённая пользователем”. А ты купи слона!» Затем программа получает новый ответ от пользователя и так до бесконечности.



#### Задачи:

- 4) С помощью бесконечного цикла и оператора `break` реализовать игру «Угадай число». Программа генерирует случайное число в диапазоне от 1 до 7 и говорит пользователю: «Угадай число от 1 до 7!». В бесконечном цикле программа считывает ответы пользователя и подсказывает ему «больше!» или «меньше!», а в случае правильного ответа цикл завершается, и выводятся сердечные поздравления победой.
- 5) Доработать программу «Угадай число» так, чтобы она не заканчивалась, то есть, чтобы после угадывания одного числа, начиналось угадывание следующего. Увеличить диапазон с 7 до 15 и ввести ограничение количество попыток (например, 3 попытки). Если пользователь не справился за 3 попытки, вывести «GAME OVER».

Дополнительные материалы:

Программа «Угадай число» вариант-1:

```
import random
a=random.randint(1,7)
print("Угадай число от 1 до 7")
while True:
    b=int(input())
    if b==a:
        print("Верно!")
```

	<pre> break elif b&lt;a:     print("Больше!") else:     print("Меньше!") Программа «Угадай число» вариант-2: import random while True:     a=random.randint(1,15)     print("Угадай число от 1 до 15")     k=0     while True:         k=k+1         if k&gt;3:             print("GAME OVER")             break         b=int(input())         if b==a:             print("Пусть другие говорят:")             print("Главное – участие.")             print("Лишь победа дарит нам")             print("Наслаждение, счастье!")             print("Вас с победой поздравляем!")             print("От души мы вам желаем")             print("Без намёков и прикрас")             print("Побеждать ещё 100 раз!")             break         elif b&lt;a:             print("Больше!")         else:             print("Меньше!") </pre>	
<p><b>Конец урока</b> <b><u>9</u> мин</b></p>	<p>1. Учащиеся с помощью стикеров оценивают усвоение темы по методу «Лестница успеха»</p> <ul style="list-style-type: none"> <li>- Понял частично</li> <li>- Есть вопросы</li> <li>- Мне все понятно</li> </ul> <p>2. Домашнее задание - Написать программы:</p> <ol style="list-style-type: none"> <li>1) С помощью цикла while вывести повторяющуюся строчку из любой песни 25 раз.</li> <li>2) Написать программу, которая получает два целых числа A и B (<math>0 &lt; A &lt; B</math>) и выводит квадраты всех натуральных чисел в интервале от A до B.</li> <li>3) Дано целое число N (&gt;0). Используя операции деления нацело и взятия остатка от деления, вывести все его цифры, начиная с самой правой.</li> </ol>	

	<p>4) ** Напишите программу, которая бы «подбрасывала» условную монету 100 раз и сообщала, сколько раз выпал орел, а сколько — решка.</p>	
<p><b>Дифференциация – каким образом Вы планируете оказать больше поддержки? Какие задачи Вы планируете поставить перед более способными учащимися?</b></p>	<p><b>Оценивание – как Вы планируете проверить уровень усвоения материала учащимися?</b></p>	<p><b>Охрана здоровья и соблюдение техники безопасности</b></p>