

### Лабораторная работа №3

#### Организация циклов, вложенных циклов в программах.

Цель работы: научиться разрабатывать и отлаживать программы с использованием циклов for и while, находить оптимальные решения задач, содержащих циклы с известным и неизвестным числом повторений

#### Задание на лабораторную работу:

1. Изучить теоретические сведения.
2. Написать программу в соответствии с вариантом.

#### Теоретические сведения

Цикл **while** позволяет выполнить одну и ту же последовательность действий, пока проверяемое условие истинно. Условие записывается до тела цикла и проверяется до выполнения тела цикла. Как правило, цикл while используется, когда невозможно определить точное значение количества проходов исполнения цикла.

##### **Синтаксис цикла while**

```
while условие:  
    блок инструкций
```

При выполнении цикла while сначала проверяется условие. Если оно ложно, то выполнение цикла прекращается и управление передается на следующую инструкцию после тела цикла while. Если условие истинно, то выполняется инструкция, после чего условие проверяется снова и снова выполняется инструкция. Так продолжается до тех пор, пока условие будет истинно. Как только условие станет ложно, работа цикла завершится и управление передается следующей инструкции после цикла.

Цикл **for**, также называемый циклом с параметром, в языке Питон богат возможностями. В цикле for указывается переменная и множество значений, по которому будет пробегать переменная. Множество значений может быть задано списком, кортежем, строкой или диапазоном.

##### **Синтаксис цикла for**

```
for i in s:  
    инструкции
```

Функция **range** ([<начало>, ] <конец>[, <шаг>]) возвращает *диапазон* – особый объект, поддерживающий итерационный протокол.

Если параметр <начало> не указан, то по умолчанию берется значение 0.

Параметр <конец> задает конечное значение и не входит в возвращаемые значения.

Параметр <шаг> по умолчанию равен 1, может быть отрицательным.

range (1,10,1)	[1;10)	все числа от 1 до 10 (не включая 10) с шагом 1
range (2,5)	2, 3, 4	все числа от 2 до 5 (не включая 5) с шагом 1
range (5,2)		Если a>b, то цикл не будет выполнен ни разу.
range (5)	0, 1, 2, 3, 4	все числа от 0 до 5 (не включая 5) с шагом 1
range (2,15,4)	2, 6, 10, 14	все числа от 0 до 5 (не включая 5) с шагом 4

#### Пример программы нахождения суммы чисел от 1 до 10

sum = 0 n = 10 <b>for</b> i in range(1, n + 1): sum += i print(sum)	i = 1 sum=0 <b>while</b> i <= 10: sum+=i i += 1
---	---

print(sum)
------------

Пример: Вывести все числа от 1 до 100, кроме чисел от 5 до 10 включительно.

```
for i in range(1, 101):  
    if 4 < i < 11:  
        continue  
    print (i)
```

Пример: программа просить пользователя корректно ввести число, пока он его не введет, и определяет его четность

```
n = input("Введите целое число: ")  
  
while type(n) != int:  
    try:  
        n = int(n)  
    except ValueError:  
        print("Неправильно ввели!")  
        n = input("Введите целое число: ")  
  
if n % 2 == 0:  
    print("Четное")  
else:  
    print("Нечетное")
```

Примечание. В выражении `type(n) != int` с помощью функции `type()` проверяется тип переменной `n`. Если он не равен `int`, т. е. значение `n` не является целым числом, а является в данном случае строкой, то выражение возвращает истину. Если же тип `n` равен `int`, то данное логическое выражение возвращает ложь. При первом входе в цикл тип `n` всегда строковый, т. е. он не равен `int`. Следовательно, логическое выражение возвращает истину, что позволяет зайти в тело цикла.

## Задания

### Задание 1

B1-B12: Компьютер загадывает число от 1 до 10. У пользователя три попытки отгадать. После каждой неудачной попытки компьютер сообщает меньше или больше загаданное число. В конце игры выводит загаданное число.

### Задание 2

B1-B12: Реализуйте игру «Яблоки»: Дано  $n$  яблок. Вы и компьютер ходите по очереди. За ход можно взять 1, 2 или 3 яблока. Проиграл тот, кому нечего брать. Компьютер ходит случайно (без анализа выигрышной стратегии), однако если у него есть ход, который является последним для его выигрыша, то он его совершает.

### Задание 3

- B1.** Определите количество четных элементов в последовательности, завершающейся числом 666. Т.е. пользователь вводит числа, программа подсчитывает количество четных чисел до тех пор, пока не будет введено число 666
- B2.** Определите среднее значение всех элементов последовательности, завершающейся числом 999. Т.е. пользователь вводит числа, и программа высчитает среднее арифметическое и завершит работу, когда будет введено 999
- B3.** Последовательность состоит из натуральных чисел и завершается числом 0. Определите значение наибольшего элемента последовательности.
- B4.** Пользователь вводит натуральные числа до тех пор, пока не введет 0. Определите, сколько элементов этой последовательности больше предыдущего элемента. (Натуральные числа — это числа, используемые для счета предметов или для указания порядкового номера, т.е. от 1 до  $\infty$ )
- B5.** Определите сумму всех элементов последовательности, завершающейся числом 100. Т.е. пользователь вводит числа и программа считает их сумму, пока не будет введено 100
- B6.** Программа получает на вход последовательность целых неотрицательных чисел, каждое число записано в отдельной строке. Последовательность завершается числом 0, при считывании которого программа должна закончить свою работу и вывести количество членов последовательности (не считая завершающего числа 0).
- B7.** Дано  $N$  чисел: сначала вводится число  $N$ , затем вводится ровно  $N$  целых чисел. Подсчитайте количество положительных чисел среди введенных и выведите это количество.
- B8.** По данному натуральному  $n$  выведите лесенку из  $n$  ступенек,  $i$ -я ступенька состоит из чисел от 1 до  $i$  с пробелами.
- B9.** Дано  $N$  чисел: сначала вводится число  $N$ , затем вводится ровно  $N$  целых чисел. Подсчитайте количество нулей среди введенных чисел и выведите это количество.
- B10.** Пользователь вводит последовательность отрицательных чисел, завершающуюся числом 1. Найти наибольшее отрицательное число.
- B11.** Выведите на экран квадрат из нулей и единиц, причем нули находятся только на диагонали квадрата. Всего в квадрате сто цифр.
- B12.** Определите произведение всех элементов последовательности, завершающейся числом -1. Т.е. пользователь вводит числа, и программа считает их произведение, пока не будет введено число -1.

### Задание 4

B1. Найти сумму  $\cos \frac{3}{5} + \cos \frac{5}{7} + \cos \frac{7}{9} + \dots + \cos \frac{111}{113}$ .

B2. Пользователь вводит число. Найдите количество нечетных цифр в этом числе.

B3. Напишите программу, которая считывает два числа  $a$  и  $b$ , считает и выводит среднее арифметическое всех чисел из отрезка  $[a; b]$ , которые кратны числу 3. Например, среднее арифметическое считается для чисел на отрезке  $[-5; 12]$ . Всего чисел, делящихся на 3, на этом отрезке 6: -3, 0, 3, 6, 9, 12. Их среднее арифметическое равно 4.5

- B4. Напишите программу, в которой пользователь задает четыре числа  $a$ ,  $b$ ,  $c$  и  $d$ . Программа должна вывести фрагмент таблицы умножения для всех чисел отрезка  $[a;b]$  на все числа отрезка  $[c;d]$ . Числа  $a$ ,  $b$ ,  $c$  и  $d$  являются натуральными и не превосходят 10,  $a \leq b$ ,  $c \leq d$ .

Следуйте формату вывода из примера, для разделения элементов внутри строки используйте '\t' — символ табуляции. Заметьте, что левым столбцом и верхней строкой выводятся сами числа из заданных отрезков (заголовочные столбец и строка таблицы).

Например, пользователь ввел 7,10,5,6, программа вывела

	5	6
7	35	42
8	40	48
9	45	54
10	50	60

- B5. Пользователь хотел бы снять  $X$  \$ с банкомата. За каждый успешный вывод средств банк взимает 0,25 \$. Банкомат примет транзакцию только в том случае, если  $X$  кратно 5, а на балансе счета достаточно наличных средств для выполнения транзакции снятия средств (включая банковские сборы). Рассчитать баланс счета пользователя после попытки транзакции. Пользователь вводит количество денег для снятия ( $0 < X \leq 5000$ ) и начальный баланс счета ( $0 \leq Y \leq 5000$  с двумя цифрами точности). Например,

30	42
120.00	120.00

Программа выводит баланс счета:

89.25	120.00
-------	--------

Successful Transaction

Incorrect Amount (not multiple of 5)

- B6. Вычислите значения функции  $f(x)=x^3-\cos x$  на отрезке  $[a;b]$  с шагом  $h$ . Результат представить в виде (если  $a=3$ ,  $b=10$ ,  $h=3$ ):

значение  $x$                       значение  $y$

3	...
6	...
9	...

- B7. Найти сумму всех кратных 3 или 5 чисел меньше 1000. Например, если перечислить все числа меньше 10, кратные 3 или 5, получим 3, 5, 6 и 9. Сумма этих чисел равна 23.

- B8. Каждый новый член в последовательности Фибоначчи генерируется путем добавления двух предыдущих членов (например, начиная с 1, первые 10 слагаемых будут: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...) задается натуральное число  $N$ . Определить количество членов  $a_i$  последовательности Фибоначчи, которые удовлетворяют условию  $a_i \leq N$ . Вывести полученную последовательность.

- B9. Пользователь вводит  $a$  и  $n$ . Найти 
$$p = (a+1)^2(a+2)^2 \cdot \dots \cdot (a+n)^2$$

- B10. Пользователь вводит количество образцов стали  $N$ . Напишите программу, которая определяет марку стали. Пользователь задает значения твердости, содержания углерода и прочности на разрыв. Определенная марка стали классифицируется в соответствии со следующими условиями:

Твердость должна быть больше 50.

Содержание углерода должно быть менее 0,7.

Прочность на разрыв должна быть больше 5600.

Следующие марки стали:

Оценка 10 - при соблюдении всех трех условий.

Оценка 9 при соблюдении условий (i) и (ii).

Оценка 8, если выполняются условия (ii) и (iii).

Оценка 7, если выполняются условия (i) и (iii).

Оценка 6, если выполнено только одно условие.

Оценка 5, если ни одно из трех условий не выполнено.

- B11. Пользователь вводит строку. Найти максимальное число букв из заданной строки, которое совпадает с началом английского алфавита (abcdefghijklmnopqrstuvwxyz). Например, ввели «abchjfds», получили 3.
- B12. Пользователь вводит N строк. Если длина слова больше 10 символов, то слово заменяется на аббревиатуру. Слова, не являющиеся слишком длинными, должны остаться без изменений.  
Эта аббревиатура строится следующим образом: записывается первая и последняя буква слова, а между ними — количество букв между первой и последней буквой (в десятичной системе счисления и без ведущих нулей). Например, слово «localization» запишется как «l10n», а «internationalization» как «i18n».

### Задание 5

- B1-3: Пользователь вводит строку, состоящую только из букв английского алфавита в нижнем регистре. Необходимо полностью удалить все соседние повторяющиеся буквы. После удаления дубликатов могут возникнуть новые. Удалять дубликаты необходимо до тех пор, пока не будут удалены все повторяющиеся соседние буквы.  
Например, ввели "abbaca", результат - "ca". Т.к. после удаления «bb» получается строка «aaca», ее снова необходимо очистить, т.к. образовался новый повтор «aa».
- B4-6: Напишите программу, которая позволит проверять пароль на сложность по таким критериям:  
Длина: от 5 символов.  
Буквы латинского алфавита в верхнем и нижнем регистре: A, a, Z, z.  
Цифры: от 0 до 9.  
Символы @, #, % недопустимы.  
Пользователь вводит варианты пароля до тех пор, пока они не будут соответствовать критериям.
- B7-9: Дана строка. Выбрать из строки все цифры и сложить их. Например, строка 'ab12c59p7dq', должны получить  $1+2+5+9+7=24$ .
- B10-12: Пользователь задает строку из прописных и строчных латинских букв. Программа должна делать следующее:
- удаляет все гласные буквы (гласными буквами считаются буквы "A", "O", "Y", "E", "U", "I")
  - перед каждой согласной буквой ставит символ ".",
  - все прописные согласные буквы заменяет на строчные.

### Контрольные вопросы:

1. Какие циклы существуют в Python?
2. В каких случаях используется функция range в Python?
3. В каких случаях применяется цикл while в Python?
4. Каково основное назначение цикла for?
5. Является ли цикл for циклом со счетчиком?
6. В каком случае может произойти заикливание в цикле while?
7. Что понимают под итерационной переменной? Каково ее назначение?
8. Обязательно ли наличие счетчика для цикла while?
9. Сколько раз "прокрутится" цикл в этой программе, т. е. сколько итераций он сделает?

```
total = 100
i = 0
while i < 5:
    n = int(input())
    total = total - n
    i = i + 1
print("Осталось", total)
```