# Лабораторная работа: Применение условий и циклов для создания программ на языке Python

## Цель работы:

Изучить основы условных операторов и циклов в Python и научиться применять их для решения задач и создания простых программ.

## Описание задания:

Реализовать несколько небольших программ с использованием условных операторов (`if`, `elif`, `else`) и циклов (`for`, `while`). В процессе выполнения работы студенты освоят базовые структуры управления потоком в Python.

## Инструменты:

- Среда разработки (например, PyCharm, Visual Studio Code, Jupyter Notebook)
- Установленный интерпретатор Python

## Ход работы:

### 1. Программа проверки четности числа:

Напишите программу, которая запрашивает у пользователя число и проверяет, является ли оно четным или нечетным, используя условный оператор `if`.
```python
number = int(input("Введите число: "))
if number % 2 == 0:
 print("Число четное.")
else:
 print("Число нечетное.")
```

### 2. Программа определения наибольшего из трех чисел:

Запросите у пользователя три числа и найдите наибольшее из них с помощью вложенных условий.
```python
a = int(input("Введите первое число: "))
b = int(input("Введите второе число: "))
c = int(input("Введите третье число: "))

if a >= b and a >= c:
 print("Наибольшее число:", a)
elif b >= a and b >= c:
 print("Наибольшее число:", b)
else:
 print("Наибольшее число:", c)
```

### 3. Цикл `for`: Программа вывода таблицы умножения:

Напишите программу, которая выводит таблицу умножения для числа, введенного пользователем.
```python
number = int(input("Введите число для таблицы умножения: "))
for i in range(1, 11):
 print(number, "x", i, "=", number \* i)
```

### 4. Цикл `while`: Программа угадывания числа:

Напишите программу, которая загадывает число, и пользователь должен угадать его. Используйте цикл `while`, чтобы продолжать игру, пока пользователь не угадает число.
```python
import random
secret\_number = random.randint(1, 100)
guess = None

while guess != secret\_number:
 guess = int(input("Угадайте число от 1 до 100: "))
 if guess < secret\_number:
 print("Слишком мало!")
 elif guess > secret\_number:
 print("Слишком много!")
 else:
 print("Поздравляем! Вы угадали число.")
```

### 5. Программа с вложенными циклами: Прямоугольный узор:

Используйте вложенные циклы для вывода прямоугольного узора звездочек.
```python
rows = int(input("Введите количество строк: "))
columns = int(input("Введите количество столбцов: "))

for i in range(rows):
 for j in range(columns):
 print("\*", end=" ")
 print()
```

## Контрольные вопросы:

- Что такое условные операторы, и как они помогают в управлении потоком программы?

- В чем разница между циклами `for` и `while`?

- Что такое вложенные циклы, и в каких случаях они могут быть полезны?

- Как можно выйти из бесконечного цикла, если в коде не предусмотрены условия для остановки?

## Дополнительное задание:

Напишите программу, которая запрашивает у пользователя строку и выводит только уникальные символы из этой строки, используя условия и циклы для обработки символов.