

## Практическая работа №13

**Тема:** Составление программ с применением множеств в IDE PyCharm Community.

**Цель:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с использованием с матриц в IDE PyCharm Community.

**Тип алгоритма:** Последовательный

**Постановка задачи:**

### Задача 1.

1. Сгенерировать матрицу, в которой элементы больше 10 заменяются на 0.

```
#1. Сгенерировать матрицу, в которой элементы больше 10 заменяются на 0.
```

```
# импортируем модуль random для генерации случайных чисел
import random
```

```
# определяем размеры матрицы
rows = 3
cols = 3
```

```
# создаем пустую матрицу
matrix = []
```

```
# генерируем случайные числа и заполняем матрицу
for i in range(rows):
    row = []
    for j in range(cols):
        # генерируем случайное число от 0 до 20
        num = random.randint(0, 20)
        row.append(num)
    matrix.append(row)
```

```
# выводим исходную матрицу
print("Исходная матрица:")
for row in matrix:
    print(row)
```

```
# заменяем элементы, больше 10, на 0
for i in range(rows):
    for j in range(cols):
        if matrix[i][j] > 10:
            matrix[i][j] = 0
```

```
# выводим переделанную матрицу
print("Переделанная матрица:")
for row in matrix:
    print(row)
```

## Протокол работы программы:

Исходная матрица:

```
[1, 2, 3]
```

```
[4, 5, 6]
```

```
[7, 8, 9]
```

Измененная матрица:

```
[1, 4, 6]
```

```
[8, 5, 12]
```

```
[14, 16, 9]
```

```
Process finished with exit code 0
```

## Задача 2.

В квадратной матрице все элементы, не лежащие на главной диагонали увеличить в 2 раза.

```
# В квадратной матрице все элементы, не лежащие на главной диагонали увеличить в 2 раза.

def increase_non_diagonal(matrix):
    # Вывод исходной матрицы
    print("Исходная матрица:")
    for row in matrix:
        print(row)

    # Увеличение элементов не лежащих на главной диагонали
    for i in range(len(matrix)):
        for j in range(len(matrix)):
            if i != j:
                matrix[i][j] *= 2

    # Вывод измененной матрицы
    print("\nИзмененная матрица:")
    for row in matrix:
        print(row)

# Пример использования функции
matrix = [
    [1, 2, 3],
    [4, 5, 6],
    [7, 8, 9]
]
increase_non_diagonal(matrix)
```

## Протокол работы программы:

Исходная матрица:

```
[1, 2, 3]
```

```
[4, 5, 6]
```

```
[7, 8, 9]
```

Измененная матрица:

```
[1, 4, 6]
```

```
[8, 5, 12]
```

```
[14, 16, 9]
```

```
Process finished with exit code 0
```

**Вывод:** в процессе выполнения практического занятия закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с использованием с использованием с матриц в IDE PyCharm Community. Были использованы языковые конструкции for, if. Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.