Министерство науки и высшего образования Российской Федерации   
Федеральное государственное бюджетное образовательное   
учреждение высшего образования   
«Алтайский государственный технический   
университет им. И.И. Ползунова»

Факультет информационных технологий

Кафедра прикладной математики

Отчёт защищён с оценкой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Преподаватель Андреева А.Ю.

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2022 г.

Отчёт

по лабораторной работе № 5

**«СОЗДАНИЕ СПИСКОВ С ИСПОЛЬЗОВАНИЕ СПИСКОВЫХ ВКЛЮЧЕНИЙ И ПРОСТЕЙШАЯ ОБРАБОТКА С ИСПОЛЬЗОВАНИЕ ФУНКЦИЙ LAMBLA И MAP»**

по дисциплине «Функциональное и логическое программирование»

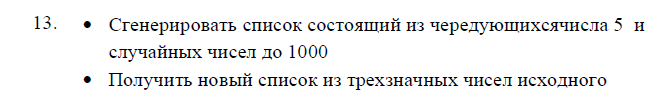
Студент группы ПИ-92 Шульпов В.М.

Преподаватель доцент, к.ф.-м.н. Андреева А.Ю.

Барнаул 2022

Номер в списке – 29

Задание 1: Используя списковые включения (list comprehension) сгенерировать одномерный список (возможно в 2-3 этапа). Выполнить обработку списка с использованием функции map и labmda.



**Решение:**

from random import randint  
  
  
def generate\_random\_list(n, num=5):  
 *"""генерация списка случайных чисел до 1000 из n элементов"""* return [randint(0, 1000) if x % 2 else num for x in range(0, n)]  
  
  
def main():  
 some\_list = generate\_random\_list(10)  
 print(some\_list)  
 new\_some\_list = list(filter(lambda x: 99 < x < 1000, some\_list))  
 print(new\_some\_list)  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

**Тесты:**

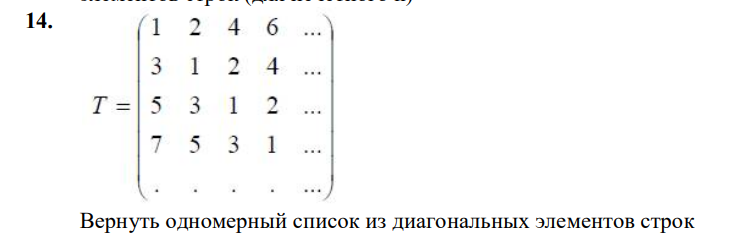
1. [5, 44, 5, 345, 5, 39, 5, 554, 5, 632]

[345, 554, 632]

1. [5, 795, 5, 161, 5, 33, 5, 306, 5, 793]

[795, 161, 306, 793]

Задание 2: Используя списковые включения (list comprehension) сгенерировать двумерный список (возможно в 2-3 этапа) порядка n (для некоторых заданий порядок должен быть нечетным) и выполнить его обработку с использованием функции lambda.



**Решение:**

from random import randint  
  
  
def generate\_matrix(n):  
 *"""генерация матрицы n порядка с заполнением по диагонялм начиная от центра"""* return [[abs(i - j) \* 2 if i < j else abs(i - j) \* 2 + 1 for j in range(0, n)] for i in range(0, n)]  
  
  
def main():  
 n = 4  
 some\_list = generate\_matrix(n)

for row in some\_list:  
 print(row)

new\_some\_list = list(filter(lambda x: x != False, [some\_list[i][j] if i == j or i == n - j - 1 else False for i in range(0, n) for j in range(0, n)]))

print(new\_some\_list)  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

**Тесты:**

