Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

|  |  |
| --- | --- |
| Факультет | инженерно-экономический |
|  |  |
| Кафедра | экономической информатики |

ОТЧЕТ

по лабораторной работе №2

на тему

**ФУНКЦИИ**

Вариант 24

Выполнил студент гр. 472302

Г.А. Сидоренко

Проверил А.С. Купрейчик

Минск 2025

**ЦЕЛЬ РАБОТЫ:**

Освоить основы создания и использования функций, области видимости переменных и применение встроенных инструментов программирования в языке Python.

**ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Функции в Python представляют собой объекты первого класса, которые могут принимать аргументы, выполнять операции и возвращать результат. Они определяются с помощью инструкции def, могут быть именными или анонимными (лямбда-функции). Важным аспектом является область видимости переменных: локальная, глобальная и нелокальная, которые определяют доступность переменных в разных частях программы. Функции могут возвращать как одно значение, так и несколько, используя кортежи, а также поддерживают различные типы аргументов: позиционные, именованные, обязательные и необязательные, а также переменное число аргументов через \*args и \*\*kwargs.

**ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ**

1. Написать функцию, которая считает сколько слагаемых должно быть сумме 1+1/2+1/3+1/4+...+1/n, чтобы эта сумма оказалась больше 4. – 2 балла;
2. Напишите функцию, которая будет принимать один аргумент. Если в функцию передаётся список, удалить все повторяющиеся элементы. Найти в новом списке сумму после первого положительного элемента. Если словарь – отсортировать в порядке убывания по значению. Число – определить простое, или нет. Строка – каждый символ перевести в соответствующий ему код из таблицы символов Unicode. Сделать проверку со всеми этими случаями. – 4 балла;
3. Найти в матрице первую строку, все элементы которой отрицательны. Найти сумму элементов этой строки. – 2 балла;
4. Напишите программу, демонстрирующую работу try\except\finally– 2 балла.

Таблица 1. Код решения задания 1

|  |
| --- |
| from itertools import count  def count\_elements(goal = 4):  summ = 0  for el in count(1):  summ += 1/el  # print(el, summ)  if summ >= goal:  return el  return None |

Таблица 2. Код решения задания 2

|  |
| --- |
| def polyglot(arg):  if isinstance(arg, list):  new\_list = []  # не используется list(set()) потому что важен порядок чисел  for i in arg:  if i not in new\_list:  new\_list.append(i)  for i in range(len(new\_list)):  if new\_list[i] > 0:  new\_list = new\_list[i:]  print(new\_list)  return sum(new\_list)  return None  elif isinstance(arg, str):  return [ord(char) for char in arg]  elif isinstance(arg, int):  return is\_prime(arg)  elif isinstance(arg, dict):  return sorted(arg, key=arg.get, reverse=True)  else :  return None  def polyglot\_test():  while True:  print("\nВыберите тип данных для тестирования:")  print("1. Список (list)")  print("2. Строка (str)")  print("3. Целое число (int)")  print("4. Словарь (dict)")  print("5. Выход")  choice = input("Введите номер варианта (1-5): ")  if choice == '1':  print("Тестирование со списком ")  try:  user\_input = input("Введите числа через пробел: ")  numbers = [int(x.strip()) for x in user\_input.split()]  print(f"Входные данные: {numbers}")  result = polyglot(numbers)  print(f"Результат: {result}")  except ValueError:  print("Ошибка: введите корректные числа!")  elif choice == '2':  print("Тестирование со строкой")  user\_input = input("Введите строку: ")  print(f"Входные данные: '{user\_input}'")  result = polyglot(user\_input)  print(f"Коды символов: {result}")  elif choice == '3':  print("Тестирование с целым числом ")  try:  user\_input = int(input("Введите целое число: "))  print(f"Входные данные: {user\_input}")  result = polyglot(user\_input)  print(f"Является ли простым числом: {result}")  except ValueError:  print("Ошибка: введите корректное целое число!")  elif choice == '4':  print("Тестирование со словарем ")  try:  print("Введите пары ключ-значение в формате: ключ1:значение1,ключ2:значение2")  user\_input = input("Например: a:5,b:2,c:8,d:1: ")  pairs = [pair.strip() for pair in user\_input.split(',')]  dictionary = {}  for pair in pairs:  if ':' in pair:  key, value = pair.split(':', 1)  dictionary[key.strip()] = int(value.strip())  print(f"Входные данные: {dictionary}")  result = polyglot(dictionary)  print(f"Ключи, отсортированные по значениям (по убыванию): {result}")  except ValueError:  print("Ошибка: введите корректные пары ключ-значение!")  elif choice == '5':  break  else:  print("Неверный выбор. Пожалуйста, введите число от 1 до 5.")  def is\_prime(n: int):  if n < 2:  return False  for i in range(2, n // 2):  if n % i == 0:  return False  return True |

Таблица 3. Код решения задания 3

|  |
| --- |
| def create\_random\_matrix(rows, cols):  matrix = []  for i in range(rows):  matrix.append([])  for j in range(cols):  matrix[i].append(random.randint(-10, 4))  return matrix  def neg\_row\_sum(matrix):  index = -1  for row in matrix:  neg = True  for num in row:  if num >= 0:  neg = False  break  if neg:  index = matrix.index(row)  break  if index != -1:  return index,sum(matrix[index])  return None, None |

Таблица 4. Код решения задания 4

|  |
| --- |
| def divide(a, b):  result = 0  try:  result = a / b  except ZeroDivisionError:  print("Деление на ноль запрещено")  finally:  print("Результат должен быть всегда")  print(result) |

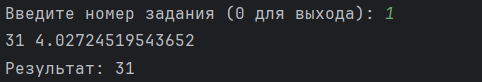


Рисунок 1. Задание 1

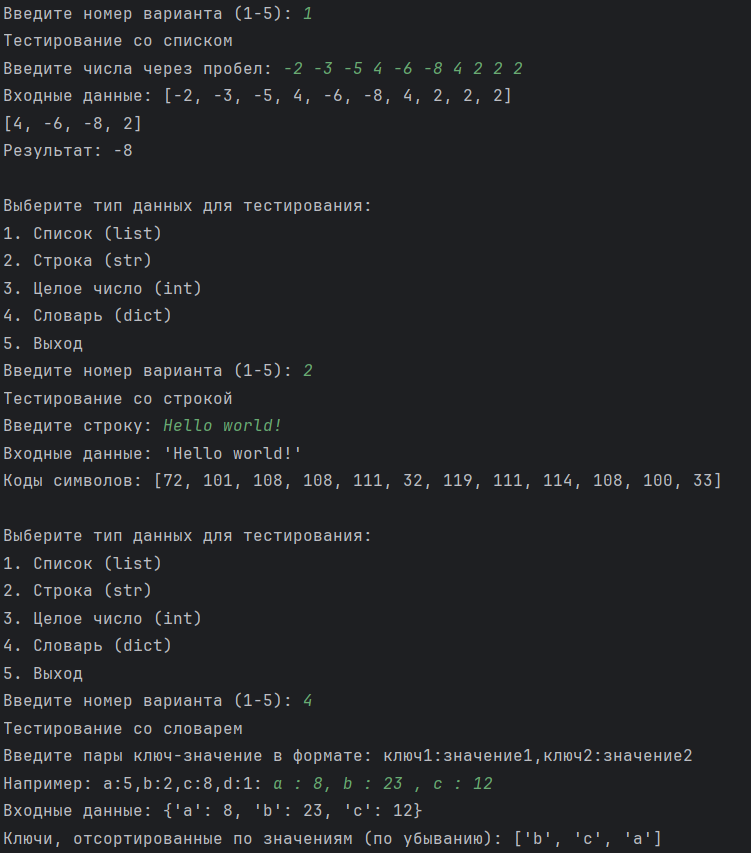


Рисунок 2. Задание 2

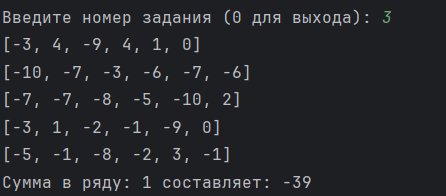


Рисунок 3. Задание 3

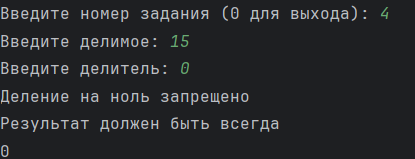


Рисунок 4. Задание 4

**ВЫВОД:** в результате выполнения лабораторной работы были получены навыки создания и использования функций в Python, изучены их основные возможности и особенности. Освоены принципы работы с областями видимости, возврата значений, а также применение встроенных функций и модулей для решения задач. Это позволяет повысить модульность и читаемость кода, упростить его повторное использование и дальнейшую модификацию.