|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| voenmeh | МИНОБРНАУКИ РОССИИ  федеральное государственное бюджетное образовательное учреждение  высшего образования  **«Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова»**  **(БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова»)** | | | | |
|  | | | | |
| Факультет | |  | О |  | Естественнонаучный |
|  | |  | Шифр |  | наименование |
| Кафедра | |  | О7 |  | Информационные системы и программная инженерия |
|  | |  | шифр |  | наименование |
| Дисциплина | |  | Прототипирование ПО | | |

ЛАБОРАТОРАЯ РАБОТА №2

|  |
| --- |
| «Функции» |
| Вариант 14 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Выполнил студент группы | | |  | И582 |
| Зиновьев Е.В. | | | | |
| Фамилия И.О. | | | | |
| **ПРЕПОДАВАТЕЛЬ** | | | | |
| Магомедов И. Н. |  |  | | |
| Фамилия И.О. Подпись | | | | |

Санкт-Петербург

2021

**Общее задание**

Переработать первую практическую, таким образом чтобы для каждого задание из первой работы была своя функция, добавить функцию проверки ввода. Параметрами функций являются используемые переменные. У переменных в функциях должны быть указаны значения по умолчанию. Функции можно вызывать как с параметрами, так и без них.

**Задание по варианту**

Функция принимает N количество параметров, четыре из которых именованные (x\_b, y\_b, x\_e и y\_e), которые по умолчанию равны 0. Функция вычисляет расстояние на плоскости между введёнными точками. Считать, что x\_b, y\_b, x\_e и y\_e – это первая прямая, следующую точку соединяем с концом предыдущей прямой.

**Задание 1**

На рисунке 1 представлен результат работы программы.

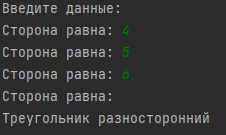


Рисунок 1 - Результат работы программы

Исходный текст программы:

def checkSide(cin = "1"):  
 if not cin.isnumeric() or float(cin) == 0:  
 print("Ошибка ввода")  
 return False  
 else:  
 return True  
  
def checkNum(cin='', numEnter = 0):  
 if cin == '' and numEnter < 3:  
 print("Введите больше цифр")  
 return False  
 else:  
 return True  
  
def checkTriangle(num1 = 4,num2 = 5, num3 = 6):  
 if not (num1 + num2) > num3 or not (num2 + num3) > num1 or not (num1 + num3) > num2:  
 print("Введенные данные некорректны")  
 return False  
 else:  
 return True  
  
def main():  
 num1 = 0  
 num2 = 0  
 num3 = 0  
 numEnter = 0  
  
 print("Введите данные: ")  
 while (1):  
 cin = input("Сторона равна: ")  
 if not checkNum(cin, numEnter):  
 continue  
 if cin == '':  
 break  
 else:  
 if not checkSide(cin):  
 continue  
 else:  
 numEnter += 1  
 num1 = num2  
 num2 = num3  
 num3 = float(cin)  
  
 if not checkTriangle(num1, num2, num3):  
 return  
  
 if num1 == num2 == num3:  
 print("Треугольник равносторонний")  
 else:  
 if num1 == num2 or num2 == num3 or num1 == num3:  
 print("Треугольник равнобедренный")  
 else:  
 print("Треугольник разносторонний")  
  
main()

**Задание 2**

На рисунке 2 представлен результат работы программы.

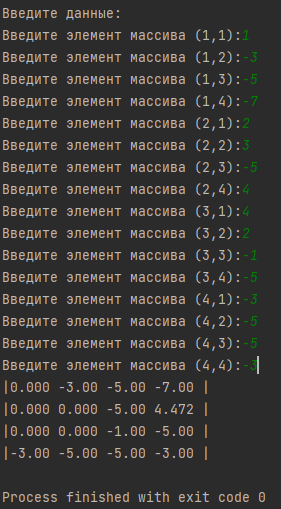


Рисунок 2 - Результат работы программы

Исходный текст программы:

import math  
def cinCheck(cin = "1"):  
 if float(cin) >= 0 and not cin.isnumeric():  
 print("Ошибка ввода")  
 return False  
 else:  
 return True  
  
def inCheck(i = 0,j = 0):  
 while (1):  
 cin = input(f"Введите элемент массива ({i + 1},{j + 1}):")  
 if not cinCheck(cin):  
 continue  
 else:  
 return cin  
  
def main():  
 sum = 0  
 error = False  
 A = [[0, 0, 0, 0],  
 [0, 0, 0, 0],  
 [0, 0, 0, 0],  
 [0, 0, 0, 0]]  
 maxNew = [0, 0, 0]  
  
 print("Введите данные:")  
 for i in range(0, 4):  
 for j in range(0, 4):  
 cin = float(inCheck(i, j))  
 if i == j:  
 sum += pow(cin, 2)  
 if i == 0 and j == 4:  
 maxNew[0] = i  
 maxNew[1] = j  
 maxNew[2] = cin  
  
 if (i + j) == 3 and cin > maxNew[2]:  
 maxNew[0] = i  
 maxNew[1] = j  
 maxNew[2] = cin  
  
 A[i][j] = float(cin)  
  
 sum = pow(sum, 0.5)  
 A[maxNew[0]][maxNew[1]] = sum  
  
 for i in range(0, 4):  
 str = '|'  
 for j in range(0, 4):  
 if A[i][j] > 0 and not (i == maxNew[0] and j == maxNew[1]):  
 A[i][j] = 0  
 if A[i][j] < 0:  
 str += f"{A[i][j]:.{2}f} "  
 else:  
 str += f"{A[i][j]:.{3}f} "  
 str += '|'  
 print(str)  
  
main()

**Задание 3**

На рисунках 3,4,5 представлен результат работы программы.

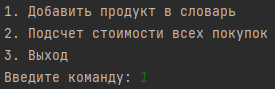


Рисунок 3 - Стартовое меню

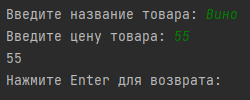


Рисунок 4 - Добавление продукта



Рисунок 5 - Подсчет общей стоимости

Исходный текст программы

def checkList(d = {"Яйца": 79, "Молоко": 67}, name = "Яйца"):  
 for element in d:  
 if element == name:  
 print("Данный товар существует")  
 return False  
 return True  
  
def checkCost(cost = "1"):  
 if not cost.isnumeric():  
 print("Введите корректную цену")  
 return False  
 else:  
 return True  
  
def getCommandMenu():  
 while (1):  
 command = float(input("Введите команду: "))  
 if not (command == 1 or command == 2 or command == 3):  
 continue  
 else:  
 return command  
  
def outMenu():  
 print("1. Добавить продукт в словарь")  
 print("2. Подсчет стоимости всех покупок")  
 print("3. Выход")  
  
def outNum(d = {"Яйца": 79, "Молоко": 67}):  
 sum = 0  
 for element in d:  
 sum += d[element]  
 print('\n' \* 100)  
 print(f"Сумма равна: {sum}")  
 while(1):  
 command = input("Нажмите Enter для возврата: ")  
 if command == '':  
 print('\n' \* 100)  
 return  
  
def addProd(d = {"Яйца": 79, "Молоко": 67}):  
 print('\n' \* 100)  
 nameExist = False  
 while (1):  
 name = input("Введите название товара: ")  
 if not checkList(d, name):  
 continue  
 else:  
 break  
 while(1):  
 cost = input("Введите цену товара: ")  
 if not checkCost(cost):  
 continue  
 else:  
 break  
  
 d[f"{name}"] = int(cost)  
  
 while (1):  
 command = input("Нажмите Enter для возврата: ")  
 if command == '':  
 print('\n' \* 100)  
 return d  
  
def main():  
 d = {"Яйца": 79, "Молоко": 67, "Яблоко": 22, "Банан": 19, "Помидор": 33, "Огурец": 31, "Пельмени": 137,  
 "Кетчуп": 59, "Майонез": 45, "Вода": 44}  
  
 while (1):  
 outMenu()  
 command = getCommandMenu()  
 print('\n' \* 100)  
 if command == 1:  
 d = addProd(d)  
 if command == 2:  
 outNum(d)  
 if command == 3:  
 return  
  
main()

**Задание 4**

На рисунке 6 представлен результат работы программы.



Рисунок 6 – Вывод длины

Исходный текст программы:

import math  
  
  
def checkNum(n = 2):  
 if n % 2 or not n:  
 print("Ошибка")  
 return False  
 else:  
 return True  
  
  
def checkN(n="1"):  
 try:  
 float(n)  
 return True  
 except ValueError:  
 print("Ошибка ввода")  
 return False  
  
  
def print\_len(len = 0):  
 print("Длина равна: %.2f" % len)  
  
  
  
def getLen(prevX = 0, prevY = 0, curX = 0, curY = 0):  
 sum = pow(curX - prevX, 2) + pow(curY - prevY, 2)  
 return pow(sum, 0.5)  
  
  
def len(x\_b = 0, y\_b = 0, x\_e = 0, y\_e = 0, \*points):  
 num = 0  
 for n in points:  
 if not checkN(n):  
 return  
 else:  
 num += 1  
 if not checkNum(num):  
 return  
  
 prevX = x\_b  
 prevY = y\_b  
 curX = x\_e  
 curY = y\_e  
  
 len = getLen(prevX, prevY, curX, curY)  
  
 if not num:  
 print\_len(len)  
 return  
  
 for i in range(num):  
 if (i + 2) % 2 == 1:  
 continue  
 if i == (num-1):  
 break  
 else:  
 prevX = curX  
 prevY = curY  
 curX = points[i]  
 curY = points[i+1]  
  
 len = getLen(prevX, prevY, curX, curY)  
  
 print\_len(len)  
  
  
len(1, 2, 3, 4, 5, 6, 15, 15)