Лабораторная работа №5

Функции.

Вариант 5(Соколова), 2(Нефёдова).

Задания

1. Реализовать примеры определения функций, листинг которых приведен в теоретической части.

Листинг 1.

|  |
| --- |
| *МЫ НЕ ПОНЯЛИ, ЧТО ВЫ ИМЕЛИ ВВИДУ ПОД РЕАЛИЗАЦИЕЙ ПРИМЕРОВ* |

1. Вычислить значение выражения f(x) в точке x, значение которой ввести с клавиатуры. Выражение задать функцией. Варианты приведены в таблице 1 (номер варианта согласовать с преподавателем)

Листинг 2.

|  |
| --- |
| import math  def f5(n):  return x\*\*3-12\*x-8  x = float(input())  print(f5(x)) |

|  |
| --- |
| import math  def f2(n):  return x\*\*4 – 5\*x – 1  x = float(input())  print(f2(x)) |

1. Вычислить значение f(x, y) в точке (x, y)значение которой ввести с клавиатуры. Выражение задать функцией. Варианты приведены в таблице 2 (номер варианта согласовать с преподавателем).

Листинг 3.

|  |
| --- |
| import math  def f1(a, b):  return math.sin(x) + 2\*y - 2 def f2(a,b):  return math.cos(y-1) + x - 0.7  x = float(input()) y = float(input()) print(f1(x, y)) print(f2(x, y)) |

|  |
| --- |
| import math  def f1(a, b):  return math.sin(y) + 2\*x – 2  def f2(a, b):  return y + math.cos(x-1) – 0.7  x = float(input())  y = float(input())  print(f1(x, y))  print(f2(x, y)) |

1. Решить задачу, определив дополнительно функцию. Варианты приведены в таблице 3 (номер варианта согласовать с преподавателем)

Листинг 4.

|  |
| --- |
| import math  is\_simple = lambda n: not [x for x in range(2, n) if not n % x] print(\*[[x, x + 2] for x in range(1, 201) if is\_simple(x) and is\_simple(x + 2)]) |

|  |
| --- |
| import math  a = int(input())  def f4(a):  print(\*filter(lambda x: (abs(x) \*\* 0.5) % 1 == 0, a))  print(f4(a)) |

1. Дан список целых чисел. Написать функцию, возвращающую… (продолжение условия по варианту в таблице 4). Реализовать двумя способами: 1 – функция принимает на вход список, 2 – функция принимает на вход переменное число параметров.

Листинг 5.

|  |
| --- |
| from random import randint from statistics import stdev  def f11(n):  if a > 0:  print(min(a))  print(stdev(a)) def f12(\*args):  if a > 0:  print(min(a))  print(stdev(a))  a=[randint(1, 100) for i in range(10)]  print(a) f11(a) f12(a) |

|  |
| --- |
| from random import randint  from statistics import mean  def f11(n):  print(sum(map(int, a.split())))  print(mean(a))  def f12(\*args):  print(sum(map(int, a.split())))  print(mean(a))  a = randit(1, 100) for i in range(10)]  print(a)  f11(a)  f12(a) |

1. Написать рекурсивную функцию, определяющую, является ли заданное натурального числа из десятичной системы счисления в N-ричную. Значение N в основной программе вводится с клавиатуры (2<=N<=16).

Листинг 6.

|  |
| --- |
| import math  def f(N, ch):  n1=''  digits="0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ"  while ch > 0:  i = ch % N  n1 = digits[i] + n1  ch //= N  return n1  N = int(input('Введите систему счисленя:')) ch = int(input('Введите число')) print(f(N, ch)) |

1. Написать рекурсивную функцию, определяющую, является ли заданное натуральное число простым (простым называется натуральное число, большее 1, не имеющее других делителей, кроме единицы и самого себя)

Листинг 7.

|  |
| --- |
| import math  def prost(d1):  if d1 > d2:  return 0  if N % d1 == 0:  return 1  else:  return prost(d1+1)  N = int(input()) d2 = int(N\*\*0.5) d1 = 2  if N == 1:  print('нет') elif prost(d1) == 0:  print('да') else:  print('нет') |

1. Дан список вещественных чисел. Используя lambda-функцию и функцию map выведите список остатков от деления этих чисел на 7.

Листинг 8.

|  |
| --- |
| import math  from random import randint  a=[randint(1, 100) for i in range(10)] new\_list = list(map(lambda x: x % 7, a)) print(a) print(new\_list) |

1. Дан список имен детей: [‘катя’, ‘маша’, ‘таня’, ‘саша’]. Сделайте так, чтобы имена начинались с заглавной буквы.

Листинг 9.

|  |
| --- |
| import math  names = str(['катя', 'маша', 'таня', 'саша']) print(names.title()) |

1. Попробуйте переписать следующий код через map и lambda-функцию. Он принимает список реальных имён и заменяет их прозвищами, используя надёжный метод.

Листинг 10.

|  |
| --- |
| import math  names = ['Маша', 'Петя', 'Вася']  for i in range(len(names)):  names[i] = hash(names[i]) print(names)  names = ['Маша', 'Петя', 'Вася'] new\_names = list(map(lambda x: hash((x)), names)) print(new\_names) |

1. Следующий пример считает, как часто слово «сети» встречается в списке строк:

Листинг 11.

|  |
| --- |
| import math  from functools import reduce  sentences = ['научиться плести рыболовные сети',  'обучать нейронные сети',  'паук ловит в сети мух']  count = 0 new\_sentences = reduce(lambda x, y: y+x.count('сети'), sentences)  print(new\_sentences) |

1. Дан список натуральных чисел. Вывести список чисел, кратных 7, используя lambda-функцию и filter.

Листинг 12.

|  |
| --- |
| import math  from random import randint  a = [randint(1, 100) for i in range(10)] new\_a = list(filter(lambda x: x % 7 == 0, a))  print(a) print(new\_a) |

1. Дано три списка: 1 – имена абитуриентов, 2 – баллы за ЕГЭ по математике, 3 – баллы за ЕГЭ по русском, 4 – баллы за ЕГЭ по информатике. Составить список в формате: [(‘Василий Акимович Кузнецов’, 85,42, 65), (‘Петр Николаевич Чириков’, 79,49,78, …].

Листинг 13.

|  |
| --- |
| import math  names = ['Даниил', 'Пётр', 'Василий', 'Юрий'] math = [80, 56, 60, 98] rus = [98, 70, 69, 57] inf = [75, 68, 74, 80]  res = list(zip(names, math, rus, inf)) print(res) |