

Лабораторная работа № 7.

Тема: Основы программирования (Python). Функции.

Цель: закрепить теоретические знания и применить их на практике.

Задание:

Вариант 1

1. Использование встроенной функции `map()`.

Напишите программу, которая использует функцию `map()`, чтобы преобразовать список строк (например, ['1', '2', '3']) в список целых чисел. Выведите результат.

2. Создание собственной функции для вычисления площади круга.

Напишите функцию `circle_area(radius)`, которая принимает радиус круга и возвращает его площадь. Формула: $S = \pi r^2$. Используйте модуль `math` для значения числа π .

3. Использование встроенной функции `filter()`.

Дан список чисел. Используйте функцию `filter()`, чтобы создать новый список, содержащий только положительные числа из исходного списка.

4. Создание собственной функции для проверки простого числа.

Напишите функцию `is_prime(number)`, которая принимает число и возвращает `True`, если оно простое, и `False` в противном случае.

5. Использование встроенной функции `sorted()` с ключом.

Дан список строк. Используйте функцию `sorted()`, чтобы отсортировать строки по их длине (от самой короткой к самой длинной).

6. Функция с переменным числом аргументов.

Напиши функцию `total(*args)`, которая возвращает сумму всех переданных чисел. Пример: `total(1, 2, 3, 4) → 10`.

7. Функция с необязательными именованными аргументами.

Напиши функцию `describe_person(name, age=None, city="Неизвестно")`, которая выводит информацию о человеке. Параметры `age` и `city` не обязательны.

8. Функция-фильтр с передачей другой функции в аргументе.

Напиши функцию `filter_list(lst, condition)`, которая принимает список и функцию-условие, и возвращает новый список, содержащий только элементы, удовлетворяющие условию.

Пример использования: `filter_list([1, 2, 3, 4], lambda x: x % 2 == 0) → [2, 4]`.

9. Поиск самого длинного слова.

Напиши функцию `longest_word(*words)`, которая возвращает слово с наибольшей длиной. Пример: `longest_word("кот", "медведь", "слон") → "медведь"`

10. Игра "Камень, Ножницы, Бумага".

Создайте функцию, которая реализует игру "Камень, Ножницы, Бумага". Функция должна принимать выбор игрока и случайным образом выбирать выбор компьютера. Затем определите победителя в этом раунде.

Вариант 2

1. Использование встроенной функции `zip()`.

Даны два списка: один содержит имена, другой — возрасты. Используйте функцию `zip()`, чтобы объединить их в список кортежей вида (имя, возраст).

2. Создание собственной функции для подсчета гласных букв

Напишите функцию `count_vowels(text)`, которая принимает строку и возвращает количество гласных букв в ней (гласные: а, е, и, о, у).

3. Использование встроенной функции `enumerate()`.

Дан список слов. Используйте функцию `enumerate()`, чтобы вывести слова вместе с их порядковыми номерами (индексами).

4. Создание собственной функции для вычисления факториала.

Напишите функцию `factorial(n)`, которая принимает число `n` и возвращает его факториал ($n! = n \times (n-1) \times \dots \times 1$).

5. Использование встроенной функции `reduce()`.

Используйте функцию `reduce()` из модуля `functools`, чтобы найти произведение всех чисел в списке.

6. Функции с переменным числом позиционных аргументов (`*args`).

Умножение всех чисел

Напиши функцию `multiply_all(*numbers)`, которая перемножает все переданные числа.

Пример: `multiply_all(2, 3, 4) → 24`

7. Функции с необязательными именованными аргументами (`**kwargs`).

Создание анкеты пользователя

Напиши функцию `create_profile(**kwargs)`, которая принимает произвольное количество именованных параметров (например, имя, возраст, город, профессия) и выводит их в формате:

Имя: Алексей

Возраст: 20

Город: Астана

Профессия: Студент

8. Функция-фильтр с передачей другой функции в аргументе.

Фильтрация строк по условию

Описание:

Напиши функцию `filter_strings(strings, condition)`, которая принимает список строк и функцию-условие, и возвращает новый список, содержащий только те строки, которые удовлетворяют условию.

Пример использования:

`filter_strings(["кот", "собака", "кит", "слон"], lambda s: len(s) > 3)`

Результат:

`["собака", "слон"]`

9. Объединение всех строк.

Напиши функцию `concat_strings(*args)`, которая соединяет все переданные строки в одну, добавляя пробел между ними.

Пример: `concat_strings("Привет", "мир", "!") → "Привет мир !"`

10. Игра "Виселица".

Создайте функцию для игры "Виселица", где компьютер случайным образом выбирает слово, а игрок должен угадать его, вводя буквы. У игрока есть ограниченное количество попыток.

Отчет должен содержать (см. образец):

- номер и тему лабораторной работы;
- фамилию, номер группы студента и вариант задания;
- скриншоты с исходным кодом программ и комментариями;
- скриншоты с результатами выполнения программ;
- пояснения, если необходимо;
- выводы.

Отчеты в формате **pdf** отправлять на email: **colledge20education23@gmail.com**