Лабараториялық жұмыс №5

1. Тізімде қолданылатын кемінде 5 функциядарды қолданып, программа жазып шығу. Тізім ретінде əр студент өзінің резюмесін ұсынсын. Жəне сол тізіммен жұмыс жасасын.

akzer\_list = [2, 5, 1, 6, 4]

print(len(akzer\_list))

akzer\_list.append(9)

print(akzer\_list)

soz = "mY NAME is AKZER"

print(soz.capitalize())

# Tizimnin songy elementin joiady

last\_element = akzer\_list.pop()

print(last\_element)

print(akzer\_list)

# Tizimdi osu retimen suryptaidy

akzer\_list.sort()

print(akzer\_list)

2. Көрсетілген есептерді шығарыңыз:

1. Əртүрлі топтардағы студенттерді əртүрлі пəндер бойынша секцияға қатысатын бағдарлама жазыңыз. Тізімді əртүрлі санаттарға байланысты сұрыптау қажет. Нəтижені экранда көрсетіңіз. Напишите программу, в которой предлагается вводить учащихся различных групп, посещающих секции по разным предметам. Требуется упорядочить список по различным категориям. Вывести результат на экран.

from itertools import groupby

students = []

while True:

student = {}

student['name'] = input('Имя студента:')

if student['name'] == '03':

break

student['group'] = input('Группа: ')

student['subject'] = input('Предмет: ')

students.append(student)

students.sort(key=lambda x: x['group'])

for group, students\_in\_group in groupby(students, key=lambda x: x['group']):

print(f'Группа {group}:')

for student in students\_in\_group:

print(f'\t{student["name"]}, {student["subject"]}')

2. Тізім қайтаратын функция жазып шығу. Алдын ала студенттердің пəндері жəне бағалары бар тізім құрастыр. Жəне сол тізім бойынша студенттің атын еңгізген кезде, сол студенттің бағаларын шығарып бертін болсын. Напишите программу которая возвращает список. Заранее подготовьте список предметов и оценок учащихся. Когда вы вводите имя учащегося, то должны отображаться оценки этого учащегося.

grades = {

'Аяулым': {'Математика': 4, 'Физика': 5, 'География': 3},

'Аружан': {'Математика': 3, 'Физика': 4, 'География': 5},

'Молдир': {'Математика': 5, 'Физика': 4, 'География': 4},

'Сапаргул': {'Математика': 4, 'Физика': 5, 'География': 5},

'Акзер': {'Математика': 3, 'Физика': 3, 'География': 4}

}

name = input("Введите имя учащегося: ")

if name in grades:

print(f"Оценки учащегося {name}:")

for subject, grade in grades[name].items():

print(f"{subject}: {grade}")

else:

print(f"Учащийся {name} не найден.")

3. Пайдаланушыға бүтін мəндерді сұрайтын жəне оларды тізім ретінде сақтайтын бағдарламаны жазыңыз. Мəндерді енгізудің аяқталуының көрсеткіші ретінде нөлді пайдалану керек. Содан кейін бағдарлама пайдаланушы енгізген барлық сандарды (нөлден басқа) өсу ретімен көрсетуі керек - əр жолға бір мəн. Сұрыптау үшін sort əдісін немесе sorted функцияны пайдаланыңыз. Напишите программу, которая будет запрашивать у пользователя целочисленные значения и сохранять их в виде списка. Индикатором окончания ввода значений должен служить ноль. Затем программа должна вывести на экран все введенные пользователем числа (кроме нуля) в порядке возрастания – по одному значению в строке. Используйте для сортировки либо метод sort, либо функцию sorted.

numbers = []

while True:

number = int(input("Введите целое число: "))

if number == 0:

break

numbers.append(number)

numbers.sort()

print("Введенные числа в порядке возрастания:")

for number in numbers:

print(number)

4. Алдыңғы жағдайдағыдай пайдаланушыдан бүтін сандарды сұрайтын жəне оларды тізім ретінде сақтайтын бағдарламаны жазыңыз. Нөл мəндерді енгізудің аяқталуының көрсеткіші ретінде де қызмет етуі керек. Бұл жолы енгізілген мəндерді кему ретімен көрсету қажет. Напишите программу, которая, как и в предыдущем случае, будет запрашивать у пользователя целые числа и сохранять их в виде списка. Индикатором окончания ввода значений также должен служить ноль. На этот раз необходимо вывести на экран введенные значения в порядке убывания.

numbers = []

while True:

number = int(input("Введите целое число: "))

if number == 0:

break

numbers.append(number)

numbers.sort(reverse=True)

print("Введенные числа в порядке убывания:")

for number in numbers:

print(number)

5. Бас жүлдені ұтып алу үшін лотерея билетіндегі алты нөмір келесі ұтыс ойынында 1-ден 49-ға дейінгі аралықта кездейсоқ алынған алты нөмірге сəйкес келуі керек. Билетіңізге алты санды кездейсоқ таңдайтын бағдарлама жазыңыз. Бұл сандар арасында көшірмелердің жоқтығына көз жеткізіңіз. Экранда билет нөмірлерін өсу ретімен көрсетіңіз. Для выигрыша главного приза необходимо, чтобы шесть номеров на лотерейном билете совпали с шестью числами, выпавшими случайным образом в диапазоне от 1 до 49 во время очередного тиража. Напишите программу, которая будет случайным образом подбирать шесть номеров для вашего билета. Убедитесь в том, что среди этих чисел не будет дубликатов. Выведите номера билетов на экран по возрастанию.

import random

numbers = random.sample(range(1, 50), 6)

numbers.sort()

print("Номера вашего лотерейного билета:")

for number in numbers:

print(number)

6. Оған параметр ретінде берілген тізім сұрыпталғанын (өсу немесе кему ретімен) көрсететін функцияны жазыңыз. Тізім сұрыпталған болса, функция True мəнін, ал кері жағдайда False мəнін қайтаруы керек. Негізгі бағдарламада пайдаланушыдан тізімге арналған сандар тізбегін сұраңыз, содан кейін тізім бастапқыда сұрыпталғаны туралы хабарды көрсетіңіз. Напишите функцию, показывающую, отсортирован ли переданный ей в качестве параметра список (по возрастанию или убыванию). Функция должна возвращать True, если список отсортирован, и False в противном случае. В основной программе запросите у пользователя последовательность чисел для списка, после чего выведите сообщение о том, является ли этот список отсортированным изначально.

def is\_sorted(spisok):

return spisok == sorted(spisok) or spisok == sorted(spisok, reverse=True)

spisok = input("Введите список чисел: ").split()

spisok = [int(x) for x in spisok]

if is\_sorted(spisok):

print("Список отсортирован.")

else:

print("Список не отсортирован.")