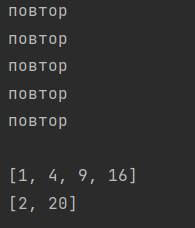
**Отчёт: Lambda функция**

def meFunction(x): #лямбда функция в функции  
 return lambda a: a \* x  
n = (lambda num1, num2: num1 + num2)(2, 3)#num1 и num2 будут подставлены значения 2 и 3  
z = (lambda y: "повтор\n" \* y)(n) # Повторяет запись ‘y’ раз  
L = [1, 2, 3, 4]  
L1 = list(map(lambda x: x\*\*2, L)) # Возводим каждый элемент массива в квадрат

print(z)  
print(L1)  
print(list(filter(lambda x: x % 2 == 0, [1, 3, 2, 5, 20, 21])))# Вывод чётных чисел

Результат:

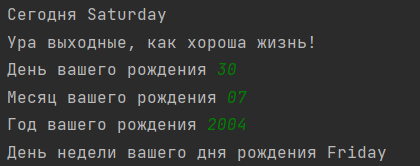


**Отчёт: Работа с датой и временем**

Код:

import datetime # импорт модуля датавремя  
d1 = datetime.datetime.today() # дата сегодня  
d1 = d1.strftime("%A") # день недели  
print("Сегодня", d1)  
if d1 == "Monday":  
 print("Рабочая неделя только началась")  
elif d1 == "Tuesday":  
 print("Начало положено, ты пережил первый день")  
elif d1 == "Wednesday":  
 print("Уже прошло 2 дня, но рано радоваться")  
elif d1 == "Wednesday":  
 print("Уже середина недели, жизнь налаживается")  
elif d1 == "Wednesday":  
 print("Осталось 2 дня, выходные уже близко")  
elif d1 == "Friday":  
 print("Последний рывок, вперёд!")  
elif d1 == "Saturday" or d1 == "Sunday":  
 print("Ура выходные, как хороша жизнь!")  
else:  
 print("Что-то тут не чисто")  
day = int(input('День вашего рождения '))  
month = int(input('Месяц вашего рождения '))  
year = int(input('Год вашего рождения '))  
date1 = datetime.date(year, month, day) # меняем местами день и год(американский стандарт)  
date2 = date1.strftime("%A")  
print("День недели вашего дня рождения", date2)

Результат:



**Работа с модулями**

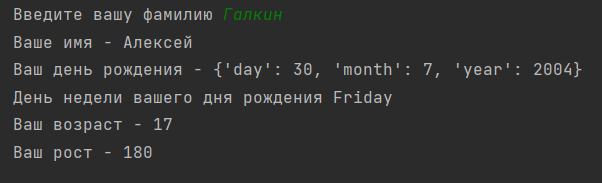
Код модуля под названием modul.py

import datetime  
vopros1 = input("Введите вашу фамилию ")  
Alexey = {  
 "name": "Алексей",  
 "surname": "Галкин",  
 "age": 17,  
 "growth": 180,  
 "birthday": {"day":30, "month": 7, "year":2004}, # Словарь в словаре  
  
}  
date = (datetime.date(Alexey["birthday"]['year'], Alexey["birthday"]['month'], Alexey["birthday"]['day'])) # Вытаскиваем из словаря значения и переводим их в дату  
day\_week = date.strftime("%A") # День недели  
Alexey["day\_week"] = day\_week # Добавляем ключ и значение в словарь  
Roma = {  
 "name": "Рома",  
 "surname": "Двойнишев",  
 "age": 17,  
 "growth": 175,  
 "birthday": {"day":13, "month": 2, "year":2004},  
  
}  
date1 = (datetime.date(Roma["birthday"]['year'], Roma["birthday"]['month'], Roma["birthday"]['day']))  
day\_week1 = date1.strftime("%A")  
Roma["day\_week"] = day\_week1

Использование модуля

import modul #импорт из файла  
vopros = modul.vopros1 #присваеваем значения переменным (переменной)  
if vopros == "Галкин":  
 name = modul.Alexey["name"] # присваеваем значения переменным (словаря)  
 age = modul.Alexey["age"]  
 growth = modul.Alexey["growth"]  
 birthday = modul.Alexey["birthday"]  
 day\_of\_the\_week\_of\_birth = modul.Alexey["day\_week"]  
if vopros == "Двойнишев":  
 name = modul.Roma["name"]  
 age = modul.Roma["age"]  
 growth = modul.Roma["growth"]  
 birthday = modul.Roma["birthday"]  
 day\_of\_the\_week\_of\_birth = modul.Roma["day\_week"]  
  
print("Ваше имя -", name, "\nВаш день рождения -", birthday)  
print("День недели вашего дня рождения", day\_of\_the\_week\_of\_birth)  
print("Ваш возраст -", age, "\nВаш рост -", growth)

**Результат:**



**Отчёт: Работа с функциями**

# №6. Составить программу перевода двоичной записи натурального числа в десятичную.

def bin\_to\_dec(chislo):  
 dlina=len(chislo)  
 chislo = [int(a) for a in str(chislo)] # Разделяем число на цифры, чтобы работать с каждой цифрой по отдельности  
 chislo\_dec=0  
 chislo\_dec\_mas = []  
 for i in range(0, dlina): # Двигаемся по циклу, пока не достигнем длины числа  
 if chislo[i] == 0: # Пропускаем вычисления с 0  
 i = i + 1  
 else:  
 chislo\_dec=(chislo[i])\*(2\*\*(dlina-i-1)) # умножаем цифру на 2 в степени разряда этого числа  
 chislo\_dec\_mas.append(chislo\_dec)  
 print(chislo[i], '\*', '2 \*\*', (dlina-i-1), '=', chislo\_dec)  
 return sum(chislo\_dec\_mas)  
b = input("Введите двоичное целое число = ")  
print("")  
print("Для того, чтобы перевести из 2-й в 10-ю \nнеобходимо умножить каждую цифру числа на 2")  
print("возведенную в степень, равную разряду цифры")  
print("А потом сложить все числа, при этом вычисления с 0 можно опустить")  
print("")  
print("Двоичное целое число",b,"соответствует десятичному числу ", bin\_to\_dec(b))

**Результат:**

Введите двоичное целое число = 101010110

Для того, чтобы перевести из 2-й в 10-ю

необходимо умножить каждую цифру числа на 2

возведенную в степень, равную разряду цифры

А потом сложить все числа, при этом вычисления с 0 можно опустить

1 \* 2 \*\* 8 = 256

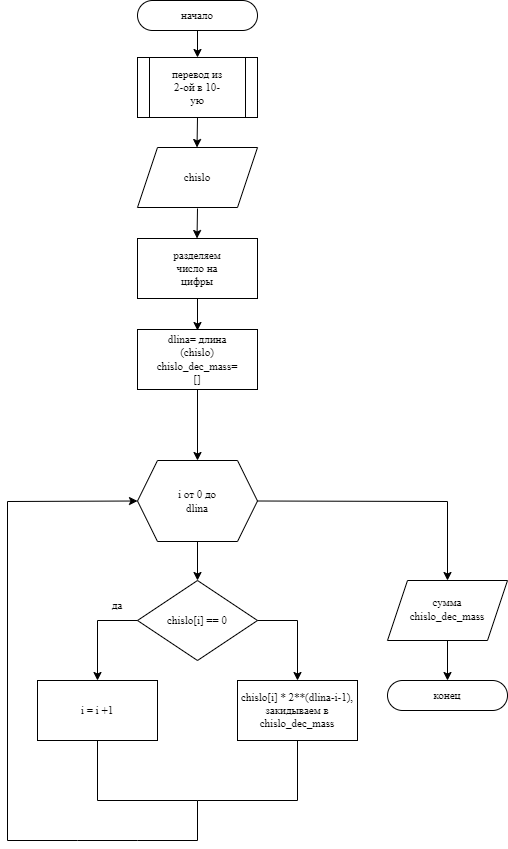
1 \* 2 \*\* 6 = 64

1 \* 2 \*\* 4 = 16

1 \* 2 \*\* 2 = 4

1 \* 2 \*\* 1 = 2

Двоичное целое число 101010110 соответствует десятичному числу 342



**Работа с массивами**

# Дан массив целых чисел. Выяснить:  
# а) Является ли s-й элемент массива положительным числом;  
# б) Является ли k-й элемент массива четным числом;  
# в) Какой элемент массива больше k или s?  
import random  
mass = [random.randrange(-1000, 1000) for i in range(11)] # Создаём рандомный массив на 10 символов  
print(mass)  
s = int(input("Введите номер элемента массива s от 0 до 10 "))   
k = int(input("Введите номер элемента массива k от 0 до 10 "))   
if mass[s] == 0: # Сравниваем s-й элемент массива с 0   
 print("а) s-й элемент", mass[s], "не является положительным числом т.к. = 0")  
else:  
 if mass[s] < 0:   
 print("а) s-й элемент", mass[s], "не является положительным числом")   
 else:   
 print("а) s-й элемент ", mass[s], "является положительным числом")   
  
if mass[k] % 2 == 0: #Вычисляем остаток   
 print("б) k-й элемент", mass[k], "является четным числом")   
else:   
 print("б) k-й элемент ", mass[k], "не является четным числом")   
  
if mass[s] == mass[k]: # Сравниваем s-й элемент массива с k-ым элементом   
 print("в) s-й элемент", mass[s], "равен элементу k", mass[k], "")   
else:   
 if mass[s] > mass[k]:   
 print("в) s-й элемент", mass[s], "больше k-го элемента", mass[k], "")   
 else:   
 print("в) k-й элемент", mass[s], "больше s-го элемента", mass[k], "")

**Результат**

