



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «ИНЖЕНЕРНЫЙ БИЗНЕС И МЕНЕДЖМЕНТ»

КАФЕДРА «ПРОМЫШЛЕННАЯ ЛОГИСТИКА» (ИБМ-3)

Рубежный контроль №1

По дисциплине:

«Парадигмы и конструкции языков программирования»

Студент ИБМЗ- 34Б

(Подпись, дата)

П. Я. Головастикова

Руководитель

(Подпись, дата)

Ю. Е. Гапанюк

2025 г.

Вариант 7. Вариант запросов А.

Код программы:

```
from operator import itemgetter

class Microprocessor:
    def __init__(self, id, model, speed, com_id):
        self.id = id
        self.model = model
        self.speed = speed
        self.com_id = com_id

class Computer:
    def __init__(self, id, name):
        self.id = id
        self.name = name

class ComputerMicroprocessor(Computer):
    def __init__(self, com_id, mic_id):
        self.com_id = com_id
        self.mic_id = mic_id

computers = [
    Computer(1, "Игровой компьютер отдел разработки"),
    Computer(2, "Серверный компьютер"),
    Computer(3, "Офисный компьютер отдел продаж"),
    Computer(4, "Рабочая станция отдел дизайна"),
    Computer(5, "Ноутбук")
]

microprocessors = [
    Microprocessor(1, "Intel Core i7", 3.8, 1),
    Microprocessor(2, "AMD Ryzen 9", 4.2, 1),
    Microprocessor(3, "Intel Xeon", 2.5, 2),
    Microprocessor(4, "Intel Core i5", 3.2, 3),
    Microprocessor(5, "AMD Ryzen 7", 3.6, 4),
    Microprocessor(6, "Intel Core i3", 2.9, 5)
]

comp_microproc = [
    ComputerMicroprocessor(1, 1),
    ComputerMicroprocessor(2, 1),
    ComputerMicroprocessor(3, 2),
    ComputerMicroprocessor(4, 3),
    ComputerMicroprocessor(5, 4),
    ComputerMicroprocessor(6, 5),
    ComputerMicroprocessor(1, 3),
    ComputerMicroprocessor(2, 4)
]

def main():
    one_to_many = [ (m.model, m.speed, c.name)
                     for m in microprocessors
                     for c in computers
                     if m.com_id == c.id]

    many_to_many_temp = [(c.name, cm.com_id, cm.mic_id)
                          for c in computers
                          for cm in comp_microproc
                          if c.id == cm.com_id]

    many_to_many = [(m.model, m.speed, com name)
```

```

        for com_name, com_id, mic_id in many_to_many_temp
        for m in microprocessors if m.id==mic_id]
print("Задание A1")
res_1 = sorted(one_to_many, key=itemgetter(2))
print(res_1)

print ('\nЗадание A2')
res_2_unsorted=[]
for c in computers:
    c_micr= list(filter(lambda i: i[2]==c.name, one_to_many))
    if len(c_micr)>0:
        c_speed= [speed for _, speed, _ in c_micr]
        c_sum_speed=sum(c_speed)
        res_2_unsorted.append((c.name,c_sum_speed))

res_2 = sorted(res_2_unsorted, key=itemgetter(1), reverse=True)
print(res_2)

print ('\nЗадание A3')
res_3={}
for c in computers:
    if 'отдел' in c.name:
        c_micr= list(filter(lambda i: i[2]==c.name, many_to_many))
        c_micr_model= [x for x,_,_ in c_micr]
        res_3[c.name] = c_micr_model
print(res_3)

if __name__ == "__main__":
    main()

```

Результат выполнения:

Задание A1

[('Intel Core i7', 3.8, 'Игровой компьютер отдел разработки'), ('AMD Ryzen 9', 4.2, 'Игровой компьютер отдел разработки'), ('Intel Core i3', 2.9, 'Ноутбук'), ('Intel Core i5', 3.2, 'Офисный компьютер отдел продаж'), ('AMD Ryzen 7', 3.6, 'Рабочая станция отдел дизайна'), ('Intel Xeon', 2.5, 'Серверный компьютер')]

Задание A2

[('Игровой компьютер отдел разработки', 8.0), ('Рабочая станция отдел дизайна', 3.6), ('Офисный компьютер отдел продаж', 3.2), ('Ноутбук', 2.9), ('Серверный компьютер', 2.5)]

Задание A3

{'Игровой компьютер отдел разработки': ['Intel Core i7', 'Intel Xeon'], 'Офисный компьютер отдел продаж': ['AMD Ryzen 9'], 'Рабочая станция отдел дизайна': ['Intel Xeon']}

