

**Московский государственный технический
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»

Отчет по рубежному контролю №1
Вариант №7 (А)

Выполнил:
студент группы ИУ5-51Б
Вардумян А.Т.

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю. Е.

Москва, 2021 г.

Задание:

1. "Компьютер" и "Микропроцессор" связаны соотношением один-ко-многим. Выведите список всех связанных микропроцессоров и компьютеров, отсортированный по компьютерам, сортировка по микропроцессорам произвольная.
2. "Компьютер" и "Микропроцессор" связаны соотношением один-ко-многим. Выведите список компьютеров с суммарной тактовой частотой микропроцессоров в каждом компьютере, отсортированный по суммарной тактовой частоте.
3. "Компьютер" и "Микропроцессор" связаны соотношением многие-ко-многим. Выведите список всех компьютеров, у которых в названии присутствует слово «компьютер», и список их микропроцессоров.

Текст программы:

```
# используется для сортировки
from operator import itemgetter

class Computer:
    """Computer"""

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

class Processor:
    """Processor"""

    def __init__(self, id, name, freq, comp_id):
        self.id = id
        self.name = name
        self.freq = freq
        self.comp_id = comp_id

class CompProc:
    """
    'Сотрудники отдела' для реализации
    связи многие-ко-многим
    """

    def __init__(self, comp_id, proc_id):
        self.comp_id = comp_id
        self.proc_id = proc_id

# Компьютеры
computers = [
    Computer(1, 'Ноутбук ACER Nitro 5 AN515-45-R9UX'),
    Computer(2, 'Компьютер ACER Aspire XC-895'),
    Computer(3, 'Компьютер ACER Aspire XC-830'),

    Computer(11, 'Ноутбук LENOVO IdeaPad S145-15API'),
```

```

        Computer(22, 'Компьютер IRU Home 615'),
        Computer(33, 'Ноутбук HP 15-dw1126ur'),
    ]

    # Процессоры
    processors = [
        Processor(1, 'AMD Ryzen 5 3600', 3.6, 1),
        Processor(2, 'AMD Athlon 3000G', 3.5, 2),
        Processor(3, 'INTEL Core i3 10100F', 3.6, 3),
        Processor(4, 'INTEL Core i5 10400F', 2.9, 3),
        Processor(5, 'AMD A6 9500', 3.5, 3),
    ]

    comp_procs = [
        CompProc(1, 1),
        CompProc(2, 2),
        CompProc(3, 3),
        CompProc(3, 4),
        CompProc(3, 5),

        CompProc(11, 1),
        CompProc(22, 2),
        CompProc(33, 3),
        CompProc(33, 4),
        CompProc(33, 5),
    ]

    def main():
        """Основная функция"""

        # Соединение данных один-ко-многим
        one_to_many = [(p.name, p.freq, c.name)
                        for c in computers
                        for p in processors
                        if p.comp_id == c.id]

        # Соединение данных многие-ко-многим
        many_to_many_temp = [(c.name, cp.comp_id, cp.proc_id)
                              for c in computers
                              for cp in comp_procs
                              if c.id == cp.comp_id]

        many_to_many = [(p.name, p.freq, comp_name)
                         for comp_name, _, proc_id in many_to_many_temp
                         for p in processors if p.id == proc_id]

        print('Задание A1')
        res_11 = sorted(one_to_many, key=itemgetter(2))
        print(res_11)

        print('\nЗадание A2')
        res_12_unsorted = []

        # Перебираем все компьютеры
        for c in computers:
            c_procs = list(filter(lambda i: i[2] == c.name, one_to_many))

```

```

    if len(c_procs) > 0:
        c_freq = [freq for _, freq, _ in c_procs]
        d_freqs_sum = sum(c_freq)
        res_12_unsorted.append((c.name, d_freqs_sum))

res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=True)
print(res_12)

print('\nЗадание A3')
res_13 = {}
# Перебираем все отделы
for c in computers:
    if 'Компьютер' in c.name:
        c_procs = list(filter(lambda i: i[2] == c.name, many_to_many))
        c_procs_names = [x for x, _, _ in c_procs]
        res_13[c.name] = c_procs_names

print(res_13)

if __name__ == '__main__':
    main()

```

Результаты программы:

arsenvardumyan@MacBook-Pro-Arsen RK1 % python rk1.py

Задание A1

```
[('INTEL Core i3 10100F', 3.6, 'Компьютер ACER Aspire XC-830'), ('INTEL Core i5 10400F', 2.9, 'Компьютер ACER Aspire XC-830'), ('AMD A6 9500', 3.5, 'Компьютер ACER Aspire XC-830'), ('AMD Athlon 3000G', 3.5, 'Компьютер ACER Aspire XC-895'), ('AMD Ryzen 5 3600', 3.6, 'Ноутбук ACER Nitro 5 AN515-45-R9UX')]
```

Задание A2

```
[('Компьютер ACER Aspire XC-830', 10.0), ('Ноутбук ACER Nitro 5 AN515-45-R9UX', 3.6), ('Компьютер ACER Aspire XC-895', 3.5)]
```

Задание A3

```
{'Компьютер ACER Aspire XC-895': ['AMD Athlon 3000G'], 'Компьютер ACER Aspire XC-830': ['INTEL Core i3 10100F', 'INTEL Core i5 10400F', 'AMD A6 9500'], 'Компьютер IRU Home 615': ['AMD Athlon 3000G']}
```