

## Текст программы:

```
# вариант 11Б (Программа Компьютер)

# используется для сортировки
from operator import itemgetter

class Prog: # program
    """Программа"""

    def __init__(self, id, name, vers, mem, comp_id):
        self.id = id
        self.name = name
        self.vers = vers # version
        self.mem = mem # memory
        self.comp_id = comp_id

class Comp: # computer
    """Компьютер"""

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

class ProgComp:
    """
    'Программы компьютера' для реализации
    связи многие-ко-многим
    """

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

# Компьютеры
comps = [
    Comp(1, 'Имя моего компьютера'),
    Comp(2, 'Мой ноутбук'),
    Comp(3, 'Рабочий компьютер'),

    Comp(11, 'имя (другого) моего компьютеров'),
    Comp(22, 'мой (другой) ноутбук'),
    Comp(33, '(другой) рабочий компьютер'),
]

# Программы
progs = [
    Prog(1, 'Chrome', '10.02.8', 262444000, 1), # ~250 MB
    Prog(1, 'Firefox', '11.22.8', 255444000, 1), # ~250 MB
    Prog(2, 'PyCharm', '12.02.8', 2600354000, 2), # ~2,5 GB
    Prog(3, 'Adobe', '6.02.8', 8100437000, 3), # ~7,6 GB
    Prog(4, 'Visual Studio', '1.02.8', 4200967000, 3), # ~4 GB
    Prog(5, 'LibreOffice', '7.02.8', 419444000, 3), # ~400 MB
]

progs_comps = [
    ProgComp(1, 1),
```

```

ProgComp(1, 2),
ProgComp(2, 3),
ProgComp(3, 4),
ProgComp(3, 5),
ProgComp(3, 6),

ProgComp(11, 1),
ProgComp(11, 2),
ProgComp(22, 3),
ProgComp(33, 4),
ProgComp(33, 5),
ProgComp(33, 6),
]

```

```
def main():
```

```
    """Основная функция"""
```

```
    # -----
```

```
    # Соединение данных один-ко-многим
```

```
    one_to_many = [(p.name, p.mem, c.name)
                    for c in comps
                    for p in progs
                    if p.comp_id == c.id]
```

```
    # Соединение данных многие-ко-многим
```

```
    many_to_many_temp = [(c.name, pc.comp_id, pc.prog_id)
                          for c in comps
                          for pc in progs_comps
                          if c.id == pc.comp_id]
```

```
    many_to_many = [(p.name, p.mem, comp_name)
                    for comp_name, com_id, prog_id in many_to_many_temp
                    for p in progs if p.id == prog_id]
```

```
    print('Задание A1')
```

```
    # «Компьютер» и «Программа» связаны соотношением один-ко-многим. Выведите список всех связанных программ и компьютеров, отсортированный по программам, сортировка по компьютерам произвольная.
```

```
    res_11 = sorted(one_to_many, key=itemgetter(0))
    print(res_11)
```

```
    print("\nЗадание A2')
```

```
    # «Компьютер» и «Программа» связаны соотношением один-ко-многим. Выведите список компьютеров с количеством программ на каждом компьютере, отсортированный по количеству программ.
```

```
    res_12_unsorted = []
    # Перебираем все компьютеры
    for c in comps:
        # Список программ компьютера
        c_progs = list(filter(lambda i: i[2] == c.name, one_to_many))
        # Если список не пустой
        if len(c_progs) > 0:
            c_prog_amount = len(c_progs)
            res_12_unsorted.append((c.name, c_prog_amount))
```

```
    # Сортировка по количеству программ
```

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

```
    print("\nЗадание A3')
```

```
    # «Компьютер» и «Программа» связаны соотношением многие-ко-многим. Выведите список всех программ, у которых название заканчивается на «е», и названия их компьютеров.
```

```
    res_13 = {}
    # Перебираем все компьютеры
```

```

for c in comps:
    # Список программ компьютера
    c_progs = list(filter(lambda i: i[2] == c.name, many_to_many))
    # Только программы с названием, заканчивающимся на "e"
    c_progs_names = [x for x, _ in c_progs if x[-1] == 'e']
    # Добавляем результат в словарь
    # ключ - компьютер, значение - список программ
    if len(c_progs_names) > 0:
        res_13[c.name] = c_progs_names
print(res_13)

```

```

if __name__ == '__main__':
    main()

```

## Результаты выполнения:

### Задание A1

```

[('Adobe', 8100437000, 'Рабочий компьютер'), ('Chrome', 262444000, 'Имя моего компьютера'),
('Firefox', 255444000, 'Имя моего компьютера'), ('LibreOffice', 419444000, 'Рабочий
компьютер'), ('PyCharm', 2600354000, 'Мой ноутбук'), ('Visual Studio', 4200967000, 'Рабочий
компьютер')]

```

### Задание A2

```

[('Рабочий компьютер', 3), ('Имя моего компьютера', 2), ('Мой ноутбук', 1)]

```

### Задание A3

```

{'Имя моего компьютера': ['Chrome'], 'Мой ноутбук': ['Adobe'], 'Рабочий компьютер':
['LibreOffice'], 'имя (другого) моего компьютеров': ['Chrome'], 'мой (другой) ноутбук':
['Adobe'], '(другой) рабочий компьютер': ['LibreOffice']}

```