

```
1  from operator import itemgetter
2
3  class Dirigent:
4      """Дирижёр"""
5      def __init__(self, id, fio, salary, orchestra_id):
6          self.id = id
7          self.fio = fio
8          self.salary = salary
9          self.orchestra_id = orchestra_id
10
11  class Orchestra:
12      """Оркестр"""
13      def __init__(self, id, name):
14          self.id = id
15          self.name = name
16
17  class DirigentOrchestra:
18      """Связь многие-ко-многим между дирижёрами и оркестрами"""
19      def __init__(self, dirigent_id, orchestra_id):
20          self.dirigent_id = dirigent_id
21          self.orchestra_id = orchestra_id
22
23
24
25  orchestras = [
26      Orchestra(1, 'Академический оркестр Москвы'),
27      Orchestra(2, 'Балтийский оркестр'),
28      Orchestra(3, 'Альфа-оркестр Санкт-Петербурга'),
29      Orchestra(4, 'Оркестр камерной музыки'),
30  ]
31
```

```

33     dirigents = [
34         Dirigent(1, 'Артамонов', 85000, 1),
35         Dirigent(2, 'Белов', 90000, 2),
36         Dirigent(3, 'Андреев', 78000, 3),
37         Dirigent(4, 'Иванов', 95000, 1),
38         Dirigent(5, 'Кузнецов', 87000, 4),
39     ]
40
41
42     dir_orch = [
43         DirigentOrchestra(1, 1),
44         DirigentOrchestra(2, 2),
45         DirigentOrchestra(3, 3),
46         DirigentOrchestra(4, 1),
47         DirigentOrchestra(5, 4),
48         DirigentOrchestra(2, 4),
49     ]
50

```

```

52 def main():
53
54     one_to_many = [(d.fio, d.salary, o.name)
55                    for o in orchestras
56                    for d in dirigents
57                    if d.orchestra_id == o.id]
58
59     many_to_many_temp = [(o.name, do.orchestra_id, do.dirigent_id)
60                           for o in orchestras
61                           for do in dir_orch
62                           if o.id == do.orchestra_id]
63
64     many_to_many = [(d.fio, d.salary, orch_name)
65                     for orch_name, orch_id, dir_id in many_to_many_temp
66                     for d in dirigents if d.id == dir_id]
67
68
69     print("Задание Г1")
70     res_1 = {}
71     for o in orchestras:
72         if o.name.startswith('А'):
73             o_dirigs = [x for x, _, orch in one_to_many if orch == o.name]
74             res_1[o.name] = o_dirigs
75     print(res_1)
76
77

```

```

77
78 print("\nЗадача Г2")
79 res_2 = [(variable) orchestras: list[Orchestra]
80 for o in orchestras:
81     o_dirigs = list(filter(lambda i: i[2] == o.name, one_to_many))
82     if o_dirigs:
83         max_salary = max([sal for _, sal, _ in o_dirigs])
84         res_2.append((o.name, max_salary))
85 res_2_sorted = sorted(res_2, key=itemgetter(1), reverse=True)
86 print(res_2_sorted)
87
88
89 print("\nЗадача Г3")
90 res_3 = sorted(many_to_many, key=itemgetter(2))
91 print(res_3)
92
93
94 if __name__ == "__main__":
95     main()
96

```

```

spiral@LAPTOP-L1VU5G9D:~/Kostylev_3SEM_25/RK1$ python3 code.py
Задача Г1
{'Академический оркестр Москвы': ['Артамонов', 'Иванов'], 'Альфа-оркестр Санкт-Петербурга': ['Андреев']}

Задача Г2
[('Академический оркестр Москвы', 95000), ('Балтийский оркестр', 90000), ('Оркестр камерной музыки', 87000), ('Альфа-оркестр Санкт-Петербурга', 78000)]

Задача Г3
[('Артамонов', 85000, 'Академический оркестр Москвы'), ('Иванов', 95000, 'Академический оркестр Москвы'), ('Андреев', 78000, 'Альфа-оркестр Санкт-Петербурга'), ('Белов', 90000, 'Балтийский оркестр'), ('Кузнецов', 87000, 'Оркестр камерной музыки')]
spiral@LAPTOP-L1VU5G9D:~/Kostylev_3SEM_25/RK1$

```