

**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»

Рубежный контроль №1.

Выполнил:
студент группы ИУ5-34Б:
Белозеров Дмитрий Сергеевич
Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Юрий Евгеньевич
Подпись и дата:

- **Вариант Г. Предметная область №2.**

Текст программы:

```
class Pupils:

    def __init__(self, id, fio, avg_mark, group_id):
        self.id = id
        self.fio = fio
        self.avg_mark = avg_mark
        self.group_id = group_id

class Group:

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

class PupGroup:

    def __init__(self, group_id, pup_id):
        self.group_id = group_id
        self.pup_id = pup_id

groups = [
    Group(1, 'А1, Первый класс'),
    Group(2, 'Б1, Третий класс'),
    Group(3, 'А2, Второй класс'),

    Group(11, 'Первый класс (прошлый год)'),
    Group(22, 'Второй класс (прошлый год)'),
    Group(33, 'Третий класс (прошлый год)'),
]

pups = [
    Pupils(1, 'Артамонов', 5.0, 1),
    Pupils(2, 'Петров', 4.7, 2),
    Pupils(3, 'Иваненко', 3.2, 3),
    Pupils(4, 'Иванов', 2.9, 3),
    Pupils(5, 'Иванин', 4.6, 3),
]

pups_groups = [
    PupGroup(1, 1),
    PupGroup(2, 2),
    PupGroup(3, 3),
    PupGroup(3, 4),
    PupGroup(3, 5),

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

def main():
    one_to_many = [(e.fio, e.avg mark, d.name)
```

```

        for d in groups
        for e in pups
        if e.group_id == d.id]

many_to_many_temp = [(d.name, ed.group_id, ed.pup_id)
                      for d in groups
                      for ed in pups_groups
                      if d.id == ed.group_id]

many_to_many = [(e.fio, e.avg_mark, dep_name)
                 for dep_name, dep_id, emp_id in many_to_many_temp
                 for e in pups if e.id == emp_id]

print('Задание Г1')
res_11 = {}
for fio, avg, name in one_to_many:
    if name[0] == 'А':
        if name in res_11:
            res_11[name].append(fio)
        else:
            res_11[name] = [fio]
print(res_11)

print('\nЗадание Г2')
res_12 = []
for g in groups:
    g_groups = list(filter(lambda i: i[2] == g.name, one_to_many))
    if len(g_groups) > 0:
        g_mark = [sal for _, sal, _ in g_groups]
        g_mark_max = max(g_mark)
        res_12.append((g.name, g_mark_max))
print(sorted(res_12, key=lambda x: x[1], reverse=True))

print('\nЗадание Г3')
res_13_unsorted = {}
for fio, avg, name in many_to_many:
    if name in res_13_unsorted:
        res_13_unsorted[name].append(fio)
    else:
        res_13_unsorted[name] = [fio]
res_13 = sorted(res_13_unsorted.items(), key=lambda x: x[0])
print(res_13)

if __name__ == '__main__':
    main()

```

Пример выполнения программы:

```

C:\Users\beloz\PycharmProjects\RK_1\venv\Scripts\python.exe C:/Users/beloz/PycharmProjects/RK_1/main.py
Задание Г1
{'А1, класс': ['Артамонов'], 'А2, класс': ['Иваненко', 'Иванов', 'Иванин']}

Задание Г2
[('А1, класс', 5.0), ('Б1, класс', 4.7), ('А2, класс', 4.6)]

Задание Г3
[('А1, класс', ['Артамонов']), ('А1, класс v1', ['Артамонов']), ('А2, класс', ['Иваненко', 'Иванов', 'Иванин']), ('А2, класс v1', ['Иваненко', 'Иванов', 'Иванин']), ('Б1, класс', ['Петров']), ('Б1, класс v1', ['Петров'])]

Process finished with exit code 0

```