```
"""
РК-1 Чиварзин А. Е. ИУ5Ц-72Б
Вариант предметной области -- 26 - Студенческая группа / Учебный курс
Вариант запросов ------------ Б
    «Группа» и «Курс» связаны соотношением один-ко-многим. Выведите список всех связанных групп и курсов, отсортированный по группам, сортировка по курсам произвольная.
    «Курс» и «Группа» связаны соотношением один-ко-многим. Выведите список курсов с количеством групп на каждом курсе, отсортированный по количеству групп.
    «Курс» и «Группа» связаны соотношением многие-ко-многим. Выведите список всех групп, у которых индекс заканчивается на «1», и номера их курсов.
"""

# используется для сортировки
from operator import itemgetter


class Course:
    def __init__(self, id_, number_, group_):
        self.id = id_
        self.number = number_
        self.group = group_  # Связь


class Group:
    def __init__(self, id_, index_, course_):
        self.id = id_
        self.index = index_
        self.course = course_  # Связь


class CourseGroup:
    """
    Связь многие ко многим
    """

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


"""
Курсы
"""
courses = [
    Course(1, '1-й', 1),
    Course(2, '2-й', 6),
    Course(3, 'ВУЦ', 1)
]

"""
Группы
"""
groups = [
    Group(1, 'ИУ5-11', 1),
    Group(2, 'ИУ5-12', 1),
    Group(3, 'ИУ5-13', 1),
    Group(4, 'ИУ5-14', 1),
    Group(5, 'ИУ5-15', 1),
    Group(6, 'ИУ5-21', 2),
    Group(7, 'ИУ5-22', 2),
    Group(8, 'ИУ5-23', 2),
    Group(9, 'ИУ5-24', 2),
    Group(10, 'ИУ5-25', 2),
```

```
    Group(11, 'ИУ1-11', 1),
]


def main():
    # Один ко многим   Курс:Группа
    one_to_many_gc = [(c.number, g.index)
                      for g in groups
                      for c in courses
                      if g.course == c.id]
    # Один ко многим   Группа:Курс
    one_to_many_cg = [(g.index, c.number)
                      for c in courses
                      for g in groups
                      if c.group == g.id]
    # Многте ко многим
    many_to_many_temp = [(g.index, c.id, c.group)
                         for g in groups
                         for c in courses
                         if g.id == c.group]
    many_to_many = [(g_index, c_id, g.course)
                    for g_index, c_id, c_group in many_to_many_temp
                    for g in groups
                    if g.id == c_id]

    print('Задание Б1')  #
######################################################################
    result_1 = sorted(one_to_many_gc, key=itemgetter(1))
    print(result_1)
    print('\nЗадание Б2')  #
######################################################################
    unsorted_result_2 = []
    # Перебираем все курсы
    for c in courses:
        # Список групп курса
        c_groups = list(filter(lambda i: i[1] == c.number, one_to_many_cg))
        # Количество групп на курсе
        c_count_g = len(c_groups)
        unsorted_result_2.append((c.number, c_count_g))
    # Сортировка
    result_2 = sorted(unsorted_result_2, key=itemgetter(1))
    print(result_2)
    print('\nЗадание Б3')  #
######################################################################
    result_3 = {}
    # Перебираем все группы
    for g in groups:
        g_course = list(filter(lambda i: i[0] == g.index, many_to_many))
        # Только индекс группы
        g_index_course = [x for x, _, _ in g_course]
        #print(g_index_course)
        if g.index[:-2:-1] == "1":
            result_3.update({(g.index, tuple(g_course))})
    print(result_3)


if __name__ == '__main__':
    main()
```

Результаты выполнения программы

```
"A:\sasha\YandexDisk\МГТУ\7-й семестр\СиТК\ДЗ\1\Ру\RK-1\Scripts\python.exe" "A:/sasha/YandexDisk/МГТУ/7-й семестр/РИП/РК/Ру/RK-1/RK.py"
Задание Б1
[('1-й', 'ИУ1-11'), ('1-й', 'ИУ5-11'), ('1-й', 'ИУ5-12'), ('1-й', 'ИУ5-13'), ('1-й', 'ИУ5-14'), ('1-й', 'ИУ5-15'), ('2-й', 'ИУ5-21'), ('2-й', 'ИУ5-22'), ('2-й', 'ИУ5-23'), ('2-й', 'ИУ5-24'), ('2-й', 'ИУ5-25')]

Задание Б2
[('1-й', 1), ('2-й', 1), ('ВУЦ', 1)]

Задание Б3
{'ИУ5-11': (('ИУ5-11', 1, 1), ('ИУ5-11', 3, 1)), 'ИУ5-21': (('ИУ5-21', 2, 1),), 'ИУ1-11': ()}

Process finished with exit code 0
```