

**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 №2
Вариант 7**

Выполнил:
студент группы ИУ5-34Б
Ковыршин Павел

Подпись и дата:

Проверил:
Гапанюк Ю.Е.

Подпись и дата:

Москва, 2021 г.

Постановка задачи.

Вариант Г.

1. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список всех отделов, у которых название начинается с буквы «А», и список работающих в них сотрудников.
2. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список отделов с максимальной зарплатой сотрудников в каждом отделе, отсортированный по максимальной зарплате.
3. «Отдел» и «Сотрудник» связаны соотношением многие-ко-многим. Выведите список всех связанных сотрудников и отделов, отсортированный по отделам, сортировка по сотрудникам произвольная.

7	Микропроцессор	Компьютер
---	----------------	-----------

- 1) Проведите рефакторинг текста программы рубежного контроля №1 таким образом, чтобы он был пригоден для модульного тестирования.
- 2) Для текста программы рубежного контроля №1 создайте модульные тесты с применением TDD - фреймворка (3 теста).

Текст программы.

```
class computer:
    id = 0

    def __init__(self, name):
        self.__id = computer.id
        computer.id += 1
        self.name = name

    def getid(self):
        return self.__id

class cmlink:
    def __init__(self, comp_id, mp_id):
        self.comp_id = comp_id
        self.mp_id = mp_id

class microprocessor:
    id = 0

    def __init__(self, name, price, comp_id):
        self.__id = microprocessor.id
        microprocessor.id += 1
        self.name = name
        self.price = price
        self.comp_id = comp_id

    def getid(self):
        return self.__id
```

```

from computer import computer
from microprocessor import microprocessor
from cmlink import cmlink

# задание 1: вывести список всех компьютеров, название которых начинается с 'A',
# и список всех микропроцессоров в них
def task1(computers, microprocessors):
    dict_base = {
        comp : [mp for mp in microprocessors if mp.comp_id == comp.getid()]
        for comp in computers
    }
    result = {}
    for comp in dict_base.keys():
        if (comp.name.lower()[0] == 'a'):
            result[comp.name] = [mp.name for mp in dict_base[comp]]
    return result

# задание 2: вывести список компьютеров и максимальную стоимость микропроцессора
# в каждом компьютере, отсортированный по убыванию стоимости
def task2(computers, microprocessors):
    dict_base = {
        comp : [mp for mp in microprocessors if mp.comp_id == comp.getid()]
        for comp in computers
    }
    result = {
        comp.name : max([mp.price for mp in dict_base[comp]])
        for comp in dict_base.keys()
    }
    result = sorted(result.items(), key= lambda x : x[1], reverse= True)
    return result

# задание 3: вывести список всех связанных пар компьютеров и микропроцессоров,
# отсортированный по компьютерам
def task3(computers, microprocessors, cmlinks):
    pairlist = [
        (computers[cmlink.comp_id].name, microprocessors[cmlink.mp_id].name)
        for cmlink in cmlinks
    ]
    return pairlist

```

```

import unittest
from computer import computer
from microprocessor import microprocessor
from cmlink import cmlink
import main

computers = [
    computer("Computer Sergey"),
    computer("Amazing Computer 3000"),
    computer("The Box")
]

microprocessors = [
    microprocessor("intel core i3", 10000, computers[0].getid()),
    microprocessor("intel core i5", 15000, computers[1].getid()),
    microprocessor("intel core i7", 30000, computers[2].getid()),
    microprocessor("intel xeon", 60000, computers[2].getid())
]

# список для третьего задания. связи в первых двух заданиях отличаются
# и никак не связаны с третьим заданием.
# в третьем задании поле id микропроцессора игнорируется
cmllinks = [
    cmlink(computers[0].getid(), microprocessors[0].getid()),
    cmlink(computers[1].getid(), microprocessors[0].getid()),
    cmlink(computers[1].getid(), microprocessors[1].getid()),
    cmlink(computers[2].getid(), microprocessors[0].getid()),
    cmlink(computers[2].getid(), microprocessors[1].getid()),
    cmlink(computers[2].getid(), microprocessors[2].getid()),
    cmlink(computers[2].getid(), microprocessors[3].getid())
]

```

```

class rk2_test(unittest.TestCase):
    def test1(self):
        self.assertEqual(main.task1(computers, microprocessors),
        {
            'Amazing Computer 3000': ['intel core i5']
        })

    def test2(self):
        self.assertEqual(main.task2(computers, microprocessors),
        [
            ('The Box', 600000),
            ('Amazing Computer 3000', 15000),
            ('Computer Sergey', 10000)
        ])

    def test3(self):
        self.assertEqual(main.task3(computers, microprocessors, cmlinks),
        [
            ('Computer Sergey', 'intel core i3'),
            ('Amazing Computer 3000', 'intel core i3'),
            ('Amazing Computer 3000', 'intel core i5'),
            ('The Box', 'intel core i3'),
            ('The Box', 'intel core i5'),
            ('The Box', 'intel core i7'),
            ('The Box', 'intel xeon'),
        ])

if __name__ == "__main__":
    unittest.main()

```

Результат выполнения программы.

```

...
-----
Ran 3 tests in 0.001s
OK

```