

Уровень сложности: сложный

Работу выполнил Волков Алексей Александрович (vk.com/relatinchik)

Для создания работы не пользовался интернетом, создавал её из-за недавнего парсинга списков поступающих в ВУЗах и наболевшего для меня «умного рейтинга». Все файлы сгенерированы случайно, по необходимости могу предоставить файлы, с помощью которых они были собраны (Совпадения с реальной жизнью случайны!).

26 Задание

Недавно все ученики сдали ЕГЭ, стали абитуриентами. Некоторые из них узнали, что Сергей Сергеевич Крылов является директором 8 института (кафедры) МАИ и решили учиться у него. Ребята подали заявление на КНИПМ в МАИ, приняли участие в конкурсе по баллам ЕГЭ. Определить количество абитуриентов, чей конкурсный балл не меньше среднего балла поступивших на направление (включая лиц, поступающих по квотам) на 5 единиц, и суммарное количество баллов по профильным предметам (математике, информатике) абитуриентов, поступивших по квотам (при условии: конкурсный балл больше максимального непроходного конкурсного балла на направление по общему конкурсу).

Участником конкурса считается каждый абитуриент в списке. Для определения своего места в рейтинге происходит сортировка в следующем порядке: по конкурсным баллам, сумме баллов ЕГЭ, баллам по математике, информатике, русскому языку, индивидуальным достижениям и статусу в конкурсе (сначала зачисляются абитуриенты со статусами: БВИ, особая квота, отдельная квота, целевая квота; потом общий конкурс). Конкурсный балл - сумма баллов абитуриента за 3 предмета ЕГЭ и баллы за индивидуальные достижения. Если абитуриент имеет статус БВИ, то его суммарный балл за ЕГЭ равен 300 (каждый предмет имеет 100 баллов за экзамен). На особую и отдельную квоту существует конкурс (не более 10% от общего количества мест на направлении), если абитуриент его не проходит, то переходить в общий конкурс.

Входные данные

В первой строке входного файла находятся два числа N — количество абитуриентов, участвующих в рейтинге (натуральное число, не превышающее 10 000) и K - количество абитуриентов, которых набирают в этом году на направление (натуральное число, не превышающее 1000). В каждой из следующих N строк находится пять чисел, разделённых пробелом: количество баллов ЕГЭ абитуриента за математику (число от 0 до 100), количество баллов ЕГЭ абитуриента за информатику (число от 0 до 100), количество баллов ЕГЭ абитуриента за русский язык (число от 0 до 100), количество баллов абитуриента за индивидуальное (число от 0 до 10), и статус (число от 1 до 5: 1 - БВИ, 2 - особая квота, 3 - отдельная квота, 4 - целевая квота, 5 - общий конкурс).

Выходные данные

Два числа: количество абитуриентов, чей конкурсный балл не меньше среднего конкурсного балла поступивших на 5 единиц, и суммарное количество баллов по математике и информатике у абитуриентов, зачисленных по квотам, чей конкурсный балл больше максимального непроходного балла по общему конкурсу.

Типовой пример рейтинга во входном файле:

15 10

3 14 85 1 2

10 62 94 8 5
79 1 8 9 1
66 98 8 5 5
34 8 44 6 5
35 16 22 8 5
79 76 98 6 1
83 100 2 7 2
23 62 68 10 3
4 94 62 0 5
60 3 9 2 5
17 10 15 3 5
90 49 34 7 5
21 4 1 5 2
5 74 92 10 4

При приведённых выше данных ответ будет: 2 347. На направление не пройдут абитуриенты с 5-ой, 6-ой, 11-ой, 12-ой, 14-ой строк примера: абитуриенты или не проходят общий конкурс, конкурсу по квоте, или только общий конкурс. Средний балл поступивших будет равен 194,5. В таком случае только абитуриенты со статусом БВИ будут иметь больше конкурсных баллов — первый ответ 2. По квотам пройдут абитуриенты с 8-ой, 9-ой и 15-ой строк. Сумма баллов по профильным предметам будет равен 347.

Пошаговое решение:

- 1) Открыть файл «hard.txt» в переменную f и записать в переменные n и k данные с первой строки файла.
- 2) В список «a» внести данные абитуриентов находящиеся в следующих n-строках файла, где первое число означает — конкурсные баллы, второе — баллы за ЕГЭ, третье — баллы за математику, четвертое — баллы за информатику, пятое — баллы за русский язык, шестое — баллы за индивидуальные достижения, седьмое — статус в конкурсном списке.
- 3) Отсортировать список «a» по условию задачи (оно понятно расписано в условии, перечитайте, если не поняли его!).
- 4) Пройтись по списку «a» и разбить его данные на списки для БВИ, квот и общему конкурсу в соответствии с условием.
- 5) Провести дополнительную сортировку для отдельных списков (**обязательно для списка «kon», ведь в него будут добавлены абитуриенты участвующие по квотам, но не прошедшие по ним, чьи баллы могут быть значительно меньше, для остальных — по желанию**).
- 6) В список «postupi» добавить всех абитуриентов, прошедших конкурс, в переменную lst записать номер, увеличенный на 1, последнего прошедшего по общему конкурсу.
- 7) Переменной «max_perexod» присвоить количество баллов последнего непрошедшего по конкурсу абитуриента.
- 8) В цикле, запущенному по количеству конкурсных мест, найти сумму конкурсных баллов всех абитуриентов в переменной «sum», а также сумму баллов экзаменов по профильным предметам у тех, кто поступил по квоте в переменную «summprof».
- 9) Найти среднюю сумму конкурсных баллов поступивших в переменную «srsum». В цикле, запущенному по количеству конкурсных мест, найти абитуриентов, чей конкурсный балл больше среднего значения на 5.
- 10) Вывести ответ: количество абитуриентов, подходящих по условию и сумму баллов по профильным предметам абитуриентов, зачисленных по квотам.

```

1  f = open('hard.txt', 'r', encoding='utf-8')
2  n, k = map(int, f.readline().split())
3  bvi, os, ot, cel, kon = [], [], [], [], []
4  a = []
5  for i in range(n):
6      mat, info, rus, idi, st = map(int, f.readline().split())
7      a.append([mat+info+rus+idi, mat+info+rus, mat, info, rus, idi, st])
8
9      a.sort(key=lambda x: (-x[0], -x[1], -x[2], -x[3], -x[4], x[5], x[-1]))
10 for i in range(n):
11     ks, s, mat, info, rus, idi, st = a[i]
12     if st == 1:
13         bvi.append([300+idi, 300, 100, 100, 100, idi, st])
14     if st == 2 and len(os) < k * 0.1:
15         os.append([ks, s, mat, info, rus, idi, st])
16     elif st == 2 and len(os) >= k * 0.1:
17         kon.append([ks, s, mat, info, rus, idi, 5])
18     if st == 3 and len(ot) < k * 0.1:
19         ot.append([ks, s, mat, info, rus, idi, st])
20     elif st == 3 and len(ot) >= k * 0.1:
21         kon.append([ks, s, mat, info, rus, idi, 5])
22     if st == 4:
23         cel.append([ks, s, mat, info, rus, idi, st])
24     if st == 5:
25         kon.append([ks, s, mat, info, rus, idi, st])

```

```

27 bvi.sort(key = lambda x: (-x[0], -x[1], -x[2], -x[3], -x[4], x[5], x[-1]))
28 os.sort(key = lambda x: (-x[0], -x[1], -x[2], -x[3], -x[4], x[5], x[-1]))
29 ot.sort(key = lambda x: (-x[0], -x[1], -x[2], -x[3], -x[4], x[5], x[-1]))
30 cel.sort(key = lambda x: (-x[0], -x[1], -x[2], -x[3], -x[4], x[5], x[-1]))
31 kon.sort(key = lambda x: (-x[0], -x[1], -x[2], -x[3], -x[4], x[5], x[-1]))
32 postupi = []
33 lst = 0
34 for i in range(len(bvi)):
35     if len(postupi) < k:
36         postupi.append(bvi[i])
37 for i in range(len(os)):
38     if len(postupi) < k:
39         postupi.append(os[i])
40 for i in range(len(ot)):
41     if len(postupi) < k:
42         postupi.append(ot[i])
43 for i in range(len(cel)):
44     if len(postupi) < k:
45         postupi.append(cel[i])
46 for i in range(len(kon)):
47     if len(postupi) < k:
48         postupi.append(kon[i])
49     lst = i+1

```

```

51 max_neproход = kon[lst][1]
52 summ = 0
53 summprof = 0
54 for i in range(k):
55     summ += postupi[i][0]
56     if postupi[i][-1] in [2, 3, 4] and postupi[i][1] > max_neproход:
57         summprof += postupi[i][2] + postupi[i][3]
58
59 srsumm = summ / k
60 count = 0
61 for i in range(k):
62     if postupi[i][0] >= 5 + srsumm:
63         count += 1
64
65 print(count, summprof)

```