

ЕДИНАДЕСЕТИ НАЦИОНАЛЕН ЕСЕНЕН ТУРНИР ПО ИНФОРМАТИКА „ДЖОН АТАНАСОВ“
Шумен, 26.11.2011 г.

Група А (11-12 клас)

Задача А2. CLUSTERING

Автор: Александър Георгиев

Ако помните задачата Sheep трябва да знаете, че Ели често сънува как разхожда стадото си с овце. А пашата на нейните животни (както в миналата задача, така и в тази) не е лесно начинание. Ели, също така, има и кучета, които охраняват стадото ѝ от вълци и крадци. Опасността за всяка овца се определя от разстоянието ѝ до най-близкото куче. Опасността за цялото стадо пък е сумата от опасностите за всяка овца.

Ако считаме поляните за сравнително полегати (а това в общия случай е така, тъй като стадото ѝ не си пада особено по катеренето), можем да представим овцете като N точки в двумерно Евклидово пространство. Ели се чуди как да разположи своите кучета-пазачи (K точки в същото пространство), така че опасността за стадото да е минимална.

Казано с други думи, дадени са ви N точки чрез своите X - и Y -координати. От вас се иска да поставите нови K точки така, че сумата от минимумите на разстоянията от всяка от дадените точки до някоя от сложените от вас е възможно най-малка. Напишете програма **clustering**, която решава този проблем колкото може по-добре.

Вход

На първия ред на стандартния вход са зададени естествените числа N и K , разделени с интервал. Всеки от следващите N реда съдържа по две цели числа X_i и Y_i – съответно координатите на i -тата овца, разделени с интервал.

Изход

На стандартния изход изведете K на брой двойки реални числа, разделени с интервал – координатите на кучетата-пазачи, форматирани с точност до шестия знак след десетичната точка. Разрешено е те да съвпадат с координатите на някоя от овцете.

Ограничения

$$1 \leq K < N \leq 1000$$

$$1 \leq K \leq 100$$

$$0 \leq X_i, Y_i \leq 10000$$

Оценяване

За всеки тест вашето решение ще получи

$$\text{round}(\min(1, (\text{author_score} / \text{your_score}))^2 * \text{test_score})$$

точки, където author_score е резултатът, намерен от авторското решение, your_score е резултатът, намерен от вашето решение, а test_score са точките за съответния тест.

Пример

Вход	Изход
7 2	1.750000 3.250000
1 2	5.000000 5.000000
1 4	
2 5	
3 2	
4 4	
5 6	
6 5	

Пояснение: Забележете, че не сте длъжни да изведете оптималното решение.