

Тренировки по алгоритмам 5.0 от Яндекса — Занятие 2 (Линейный поиск)

🕒 2 апр 2024, 06:51:30

старт: 6 мар 2024, 22:30:00

финиш: 20 мар 2024, 20:00:00

длительность: 13д. 21ч.

...

Объявления жюри

📘

Ваше участие в соревновании завершено. Вы можете дорешивать задачи и отправлять решения вне соревнования

[Положение участников](#) [Задачи](#) [Посылки](#)

Е. Амбициозная улитка

Ограничение времени	5 секунд
Ограничение памяти	256Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Домашний питомец мальчика Васи — улитка Петя. Петя обитает на бесконечном в обе стороны вертикальном столбе, который для удобства можно представить как числовую прямую. Изначально Петя находится в точке 0. Вася кормит Петю ягодами. У него есть n ягод, каждая в единственном экземпляре. Вася знает, что если утром он даст Пете ягоду с номером i , то поев и набравшись сил, за остаток дня Петя поднимется на a_i единиц вверх по столбу, но при этом за ночь, потяжелев, съедет на b_i единиц вниз. Параметры различных ягод могут совпадать.

Пете стало интересно, а как оно там, наверху, и Вася взялся ему в этом помочь. Ближайшие n дней он будет кормить Петю ягодами из своего запаса таким образом, чтобы максимальная высота, на которой побывал Петя за эти n дней была максимальной. К сожалению, Вася не умеет программировать, поэтому он попросил вас о помощи. Найдите, максимальную высоту, на которой Петя сможет побывать за эти n дней и в каком порядке Вася должен давать Пете ягоды, чтобы Петя смог её достичь!

Формат ввода

В первой строке входных данных дано число n ($1 \leq n \leq 5 \cdot 10^5$) — количество ягод у Васи. В последующих n строках описываются параметры каждой ягоды. В $i + 1$ строке дано два числа a_i и b_i ($0 \leq a_i, b_i \leq 10^9$) — то, насколько поднимется улитка за день после того, как съест i ягоду и насколько опуститься за ночь.

Формат вывода

В первой строке выходных данных выведите единственное число — максимальную высоту, которую сможет достичь Петя, если Вася будет его кормить оптимальным образом. В следующей строке выведите n различных целых чисел от 1 до n — порядок, в котором Вася должен кормить Петю (i число в строке соответствует номеру ягоды, которую Вася должен дать Пете в i день чтобы Петя смог достичь максимальной высоты).

Пример 1

Ввод 	Вывод 
3 1 5 8 2 4 4	10 2 3 1

Пример 2

Ввод 	Вывод 
2 7 6 7 4	10 2 1

Примечания

Во втором примере изначально улитка находится на высоте 0. Пусть сначала Петя накормит её второй ягодой, а затем первой. После того как она съест вторую ягоду, за день она поднимется на 7 (и окажется на высоте 7), а за ночь опустится на 4 (и окажется на высоте 3). После того как она съест первую ягоду, за день она поднимется на 7 (и окажется на высоте 10), а за ночь опустится на 6 (и окажется на высоте 4). Таким образом, максимальная высота, на которой побывает улитка при данном порядке кормления, равна 10. Нетрудно видеть, что если Петя накормит улитку сначала первой ягодой, а затем второй, то максимальная высота, на которой побывает улитка, будет меньше.

Язык

Python 3.9 (PyPy 7.3.11) ▾

Набрать здесь

Отправить файл

```
1 from collections import deque
2
3 def get_lastLeft_and_firstRight(berries):
4     max_right, first_right = float('-inf'), None
5     max_left, last_left = float('-inf'), None
6
7     total = 0
8     for i in range(len(berries)):
9         delta = berries[i][0]-berries[i][1]
10
11         if (delta <= 0) and ((berries[i][0] > max_right)):
12             max_right = berries[i][0]
13             first_right = i
14         elif (delta > 0):
15             total += berries[i][0]-berries[i][1]
16
17             if (berries[i][1] > max_left):
18                 max_left = berries[i][1]
19                 last_left = i
20
21     return last_left, first_right, total
22
23 def answer(berries):
24     left_height = 0
25     right_height = 0
26
27     last_left, first_right, total_left_height = get_lastLeft_and_firstRight(berries)
28     result = deque()
29
30     if last_left is not None:
31         result.append(last_left+1)
32         left_height = total_left_height+berries[last_left][1]
33
34     if first_right is not None:
35         result.append(first_right+1)
36         right_height = total_left_height+berries[first_right][0]
37
38     for i in range(len(berries)):
```

Отправить

📘 осталось 100 попыток

Предыдущая

Следующая

Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы	
15 мар 2024, 02:34:58	109729823	E	Python 3.9 (PyPy 7.3.11)	OK	-	0.867s	131.13Mb	-	-	отчёт
15 мар 2024, 02:29:03	109729667	E	Python 3.9 (PyPy 7.3.11)	OK	-	0.916s	114.98Mb	-	-	отчёт
15 мар 2024, 02:20:55	109729410	E	Python 3.9 (PyPy 7.3.11)	OK	-	0.944s	145.88Mb	-	-	отчёт
7 мар 2024, 06:50:24	109032460	E	Python 3.9 (PyPy 7.3.11)	OK	-	0.942s	145.89Mb	-	-	отчёт
7 мар 2024, 05:13:28	109030875	E	Python 3.9 (PyPy 7.3.11)	WA	-	215ms	28.09Mb	16	-	отчёт
7 мар 2024, 04:58:31	109030677	E	Python 3.9 (PyPy 7.3.11)	WA	-	195ms	28.09Mb	1	-	отчёт