

Задача 3. Заек

Лудият Шапкарь току що е изгубил любимия си заек (Белият заек, разбира се) някъде в последователност от N клетки и сега се опитва да го намери. Клетките са номерирани с целите числа от 1 до N . В началото заекът се намира в една неизвестна клетка от последователността и всяка секунда търсенето на Шапкаря протича по следния начин:

1. Първо, той избира една клетка от последователността и я проверява. Ще наричаме тази клетка **проверената клетка**. Ако заекът е там, търсенето приключва.
2. След това, заекът избира или да остане в същата клетка, или да скочи в съседна клетка (т.е. една клетка наляво или една клетка надясно). **Забележете, че е възможно заекът да скочи в проверената клетка, ако се е намирал в съседна на нея; това не спира търсенето!**

Решенията за това къде ще скочи заекът, са определени в зависимост от неговото настроение. В случая, заекът има следните две настроения:

1. **Изплашено настроение** – ако заекът е в това настроение, той скача в клетката, която е **по-отдалечена от проверената клетка**. Ако това е невъзможно (намира се в клетка 1 или N), той остава в същата клетка.
2. **Любопитно настроение** – когато заекът е в това настроение, той скача в клетката, която е **по-близка до проверената клетка**. Забележете, че това винаги е възможно.

Забележете, че заекът винаги скача в зависимост от последната проверена клетка и не се интересува от предишните проверени клетки.

Тъй като това е любимия заек на Шапкаря, той е много добре запознат с неговите настроения. В случая, той знае, че заекът редува настроенията си по следния начин: започва със S секунди *изплашено настроение*, последвани от C секунди *любопитно настроение*. Например, ако $S = 2$ и $C = 1$, настроенията на заека биха били следната последователност [изплашен, изплашен, любопитен, изплашен, изплашен, любопитен...].

Шапкаря е много разтревожен за своя заек и ви моли да напишете програма `rabbit.cpp`, която намира поредица от клетки, които да се проверят, така че заекът, независимо от началната си позиция, е гарантирано да бъде намерен.

Вход

На единствения ред на стандартния вход са дадени три цели числа: N , S и C , задаващи броя на клетките и описанието на настроенията на заека.

Изход

На първия ред на стандартния изход отпечатайте едно цяло число K – броя на секундите, които са необходими за търсенето на заека, използвайки вашата поредица от клетки. На втория ред, вашата програма трябва да отпечата K цели числа в интервала $[1, N]$, описващи клетките, проверени във всяка от секундите. Забележете, че в поредицата може да има повтарящи се клетки.

Оценяване

Нека K е броят на секундите във вашата поредица от клетки за търсене. Ако се опитате да проверите невалидна клетка (извън интервала $[1, N]$), или ако вашата поредица не винаги намира заека, вие ще получите 0 точки за съответния тест и съобщение *Wrong Answer*. Иначе, ако тестът носи R точки, вие ще получите pR точки, където:

- $p = 0$, ако $K > N$
- $p = 1$, ако $K \leq T$
- $p = 0.3 \left(\frac{T}{K}\right)^2$, в противен случай

Където: $T = \frac{N(S+C)}{S+2 \max(S,C)} + 3 \max(S, C)$

Ограничения

$$2 \leq N \leq 10^4$$

$$0 \leq S, C \leq 50$$

Информация за тестовите

- В 8% от тестовите $S = 0, C = 1$
- В 12% от тестовите $S = 1, C = 0$
- В 8% от тестовите $S = 1, C = 1$

Пример

Вход	Изход
12 2 1	14
	2 5 3 2 6 1 2 11 12 12 8 10 12 6

Обяснение на примера

Можете сами да се убедите, че без значение от стартовата клетка, тази поредица винаги ще намери заека. Например, нека разгледаме случая, в който заека стартира в клетка с номер 8. Търсенето ще протече по следния начин:

Секунда	Проверена клетка	Настроение на заека (Преди движението)	Движение на заека
1	2	Изплашен	8 -> 9
2	5	Изплашен	9 -> 10
3	3	Любопитен	10 -> 9
4	2	Изплашен	10 -> 11
5	6	Изплашен	11 -> 12
6	1	Любопитен	12 -> 11
7	2	Изплашен	11 -> 12
8	11	Изплашен	12 -> 12
9	12	Любопитен	Намерен

За това решение имаме $K > T$, тъй като $K = 14$ и $T = 12$.

По този начин, ще получим $p \approx 0.26$ от точките, предвидени за този тест.