

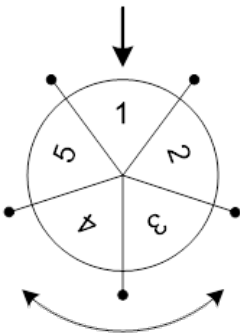
# Тренировки по алгоритмам 5.0 от Яндекса — Занятие 2 (Линейный поиск)

## Ф. Колесо Фортуны

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Развлекательный телеканал транслирует шоу «Колесо Фортуны». В процессе игры участники шоу крутят большое колесо, разделенное на сектора. В каждом секторе этого колеса записано число. После того как колесо останавливается, специальная стрелка указывает на один из секторов. Число в этом секторе определяет выигрыш игрока.

Юный участник шоу заметил, что колесо в процессе вращения замедляется из-за того, что стрелка задевает за выступы на колесе, находящиеся между секторами. Если колесо вращается с угловой скоростью  $v$  градусов в секунду, и стрелка, переходя из сектора  $X$  к следующему сектору, задевает за очередной выступ, то текущая угловая скорость движения колеса уменьшается на  $k$  градусов в секунду. При этом если  $v \leq k$ , то колесо не может преодолеть препятствие и останавливается. Стрелка в этом случае будет указывать на сектор  $X$ .



Юный участник шоу собирается вращать колесо. Зная порядок секторов на колесе, он хочет заставить колесо вращаться с такой начальной скоростью, чтобы после остановки колеса стрелка указала на как можно большее число. Колесо можно вращать в любом направлении и придавать ему начальную угловую скорость от  $a$  до  $b$  градусов в секунду.

Требуется написать программу, которая по заданному расположению чисел в секторах, минимальной и максимальной начальной угловой скорости вращения колеса и величине замедления колеса при переходе через границу секторов вычисляет максимальный выигрыш.

### Формат ввода

Первая строка входного файла содержит целое число  $n$  — количество секторов колеса ( $3 \leq n \leq 100$ ). Вторая строка входного файла содержит  $n$  положительных целых чисел, каждое из которых не превышает 1000 — числа, записанные в секторах колеса. Числа приведены в порядке следования секторов по часовой стрелке. Изначально стрелка указывает на первое число.

Третья строка содержит три целых числа:  $a$ ,  $b$  и  $k$  ( $1 \leq a \leq b \leq 10^9$ ,  $1 \leq k \leq 10^9$ ).

### Формат вывода

В выходном файле должно содержаться одно целое число — максимальный выигрыш.

## Пример 1

Ввод

5  
1 2 3 4 5  
3 5 2

Вывод

5

## Пример 2

Ввод

5  
1 2 3 4 5  
15 15 2

Вывод

4

## Пример 3

Ввод

5  
5 4 3 2 1  
2 5 2

Вывод

5

## Примечания

В первом примере возможны следующие варианты: можно придать начальную скорость колесу равную 3 или 4, что приведет к тому, что стрелка преодолеет одну границу между секторами, или придать начальную скорость равную 5, что позволит стрелке преодолеть 2 границы между секторами. В первом варианте, если закрутить колесо в одну сторону, то выигрыш получится равным 2, а если закрутить его в противоположную сторону, то — 5. Во втором варианте, если закрутить колесо в одну сторону, то выигрыш будет равным 3, а если в другую сторону, то — 4.

Во втором примере возможна только одна начальная скорость вращения колеса — 15 градусов в секунду. В этом случае при вращении колеса стрелка преодолеет семь границ между секторами. Тогда если его закрутить в одном направлении, то выигрыш составит 4, а если в противоположном направлении, то — 3.

Наконец, в третьем примере оптимальная начальная скорость вращения колеса равна 2 градусам в секунду. В этом случае стрелка вообще не сможет преодолеть границу между секторами, и выигрыш будет равен 5.

Язык Python 3.9 (PyPy 7.3.11)

Набрать здесь

Отправить файл

```
1 # считываем данные
2 n = int(input().strip()) # количество секторов колеса
3 sectors = list(map(int, input().split())) # массив значений секторов
4 a, b, k = map(int, input().split()) # мин скорость колеса, макс скорость колеса, скорость замедления
5
6 max_prize = 0
7 passed_sectors, rem = divmod(a, k)
8 if rem == 0:
9     passed_sectors -= 1
10 clockwise = passed_sectors % n
11 max_prize = max(max_prize, sectors[clockwise], sectors[- clockwise])
12
13 for cur_speed in range(a+k, min(b+1, a+n*k+1), k):
14     clockwise += 1
15     if clockwise == n:
16         clockwise = 0
17     max_prize = max(max_prize, sectors[clockwise], sectors[- clockwise])
18
19 print(max_prize)
```

Отправить

Предыдущая

Следующая