C. Саруман

|  |  |
| --- | --- |
| Ограничение времени | 4 секунды |
| Ограничение памяти | 256Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Как известно, Саруман Радужный очень любит порядок. Поэтому все полки его войска стоят друг за другом, причем каждый следующий полк содержит количество орков не меньше, чем предыдущий.

Перед тем как напасть на Хельмову Падь, Саруман решил провести несколько вылазок для разведки. Чтобы его отряды никто не заметил, он решил каждый раз отправлять несколько подряд идущих полков так, чтобы суммарное количество орков в них было равно определенному числу. Так как это всего лишь разведка, каждый полк после вылазки возвращается на свое место. Задачу выбрать нужные полки он поручил Гриме Змеиному Языку. А Грима не поскупится на вознаграждение, если вы ему поможете.

Формат ввода

В первой строке входного файла находится два целых числа: *n* (*1 ≤ n ≤ 2⋅105*) — количество полков и *m* (*1 ≤ m ≤ 2⋅105*) – количество предстоящих вылазок.

В следующей строке записано *n* чисел *ai*, где *ai* — число орков в *i*-ом полке (*1 ≤ ai ≤ 109, ai ≤ ai+1*).

Далее в *m* строках записаны запросы вида: количество полков *l* (*1 ≤ l ≤ n*), которые должны будут отправиться в эту вылазку, и суммарное количество орков в этих полках *s* (*1 ≤ s ≤ 2⋅1016*)

Формат вывода

Для каждого запроса выведите номер полка, с которого начнутся те *l*, которые необходимо отправить на вылазку. Если таких полков несколько, выведите любой. Если же так выбрать полки нельзя, выведите *-1*.

Пример

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 5 2  1 3 5 7 9  2 4  1 3 | 1  2 |

using System;

using System.IO;

internal class Program

{

public static void Main()

{

StreamReader reader = new StreamReader("input.txt");

string[] input1 = reader.ReadLine().Split();

int N = Convert.ToInt32(input1[0]);

int m = Convert.ToInt32(input1[1]);

int[] army = new int[N];

string[] input2 = reader.ReadLine().Split();

long[] sums = new long[N+1];

sums[0] = 0;

for (int i = 0; i < N; i++)

{

army[i] = Convert.ToInt32(input2[i]);

sums[i+1] = sums[i] + army[i];

}

for (int u = 0; u < m; u++)

{

string[] input3 = reader.ReadLine().Split();

int l = Convert.ToInt32(input3[0]);

long s = Convert.ToInt64(input3[1]);

int left = 0;

int right = N - l;

bool check = true;

if (army[0] > s)

{

Console.WriteLine(-1);

continue;

}

while (left < right)

{

int mid = (right + left) / 2;

if (army[mid]>s) right = mid;

else

{

long current = sums[mid + l] - sums[mid];

if (current == s)

{

Console.WriteLine(mid + 1);

check = false;

break;

}

if (current > s)

{

if (sums[mid+1] - sums[left] > s && mid - left + 1 < l)

{

left = mid;

}

else right = mid;

}

else

{

left = mid + 1;

}

}

}

if (check && sums[left + l] - sums[left] == s) Console.WriteLine(left+1);

else if (check) Console.WriteLine(-1);

}

}

}