

쌀 창고(Rice Hub)

어느 시골에, 쌀의 길 이라고 불리는 긴 직선 도로가 있다. 이 길을 따라 R 개의 논들이 있다. 각 논들은 1 이상 L 이하의 정수좌표를 갖는다. 논들은 좌표 값이 감소하지 않는 순서로 주어진다. 즉, $0 \leq i < R$ 에 대해서, 논 i 는 좌표 $X[i]$ 에 있다. $1 \leq X[0] \leq \dots \leq X[R-1] \leq L$ 라고 가정할 수 있다.

여러 개의 논이 하나의 같은 좌표에 있을 수 있음을 주의하라.

논에서 수확한 가능한 가장 많은 쌀들을 저장할 수 있는 하나의 쌀 창고를 세우려고 계획하고 있다. 논 위치와 같이, 쌀 창고의 위치는 1 이상 L 이하의 정수좌표에만 있을 수 있다. 쌀 창고는 논들이 위치하고 있는 장소를 포함하여 어떤 곳에든지 세울 수 있다.

수확의 계절에 각 논들은 정확히 트럭 한대 분량의 쌀만 생산할 수 있다. 쌀들을 창고로 옮기기 위해서 트럭 운전사들을 고용하여야 한다. 트럭 운전사는 한 트럭분량의 쌀을 창고로 옮기는데 단위 거리당 1 바트의 요금을 청구한다. 다르게 말하자면, 논으로부터 쌀을 창고로 옮기는 비용은, 논 위치 좌표 값과 창고의 좌표 값의 차이와 같다.

불행히도, 올해의 예산이 충분하지 못하여 쌀의 수송에는 최대 B 바트까지만 사용할 수 있다. 여러분이 해야 할 일은 가능한 가장 많은 쌀을 창고로 모으기 위한 창고의 위치를 결정하는 것이다.

해야 할 일(Your task)

다음의 파라미터를 갖는 `besthub(R,L,X,B)` 함수를 작성하라:

- R – 논들의 수. 논들은 0 이상 $R-1$ 이하의 번호로 나타낸다.
- L – 최대 좌표 값.
- X – 가장 작은 수부터 가장 큰 수까지 정렬된 정수의 1차원 배열.
 $0 \leq i < R$ 에 대하여, 논 i 는 $X[i]$ 에 위치한다.
- B – 예산.

여러분의 함수는 반드시 최적의 쌀 창고의 위치를 찾아야 하고, 예산 범위 내에서 쌀 창고까지 옮길 수 있는 최대 쌀의 양이 트럭 몇 대 분량 인지를 리턴 하여야 한다.

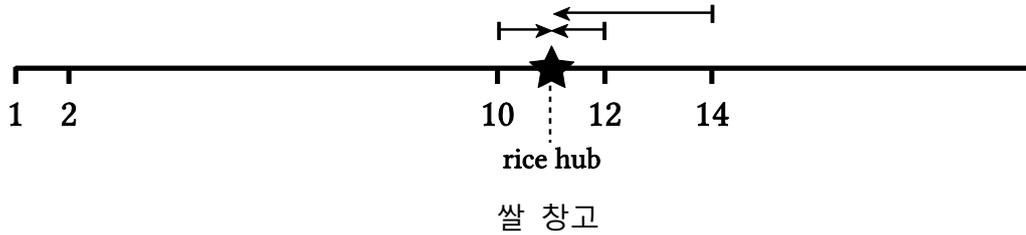
쌀을 옮기는데 허용된 예산은 매우 클 수가 있음을 주의하라. 예산은 64-비트의 정수로 주어지고, 계산 과정에서는 64-비트의 정수를 사용하도록 권장한다. C/C++ 에서는 `long long` 데이터 형을 사용하고, 파스칼에서는 `Int64` 데이터 형을 사용하라.

예제 (Example)

다음의 경우를 고려해보자. $R=5, L=20, B=6,$

```

1
2
X= 10
   12
   14
```



이 경우에는 창고의 위치에 대한 여러 개의 최적 위치가 있다: 10 이상 14 이하의 어느 정수 위치에든지 쌀 창고를 둘 수 있다. 위의 그림은 최적 위치들 중의 하나를 보여준다. 이 경우 10 과 12, 14 의 위치에 있는 논으로부터 쌀들을 옮길 수 있다. 어떤 최적의 창고 위치에 대해서든지, 쌀의 총 수송 비용은 6 바트 이하이다. 세 개보다 많은 논으로부터 쌀을 모을 수 있는 창고의 위치는 존재하지 않으므로, 이 답이 최적이고, **besthub** 은 3 을 리턴 해야 한다.

서브 태스크(Subtasks)

서브태스크 1 (17 점)

- $1 \leq R \leq 100$
 - $1 \leq L \leq 100$
 - $0 \leq B \leq 10\,000$
- ♣ 모든 논들의 좌표는 서로 다르다 (이 서브태스크 에서만).

서브태스크 3 (26 점)

- $1 \leq R \leq 5\,000$
- $1 \leq L \leq 1\,000\,000$
- $0 \leq B \leq 2\,000\,000\,000$

서브태스크 2 (25 점)

- $1 \leq R \leq 500$
- $1 \leq L \leq 10\,000$
- $0 \leq B \leq 1\,000\,000$

서브태스크 4 (32 점)

- $1 \leq R \leq 100\,000$
- $1 \leq L \leq 1\,000\,000\,000$
- $0 \leq B \leq 2\,000\,000\,000\,000\,000$

구현시 유의사항

제약조건

- CPU 제한 시간: 1 초
- 메모리 제한: 256 MB

유의사항: 스택 메모리 크기에 대한 제한은 명시되지 않는다. 스택 메모리는 전체메모리 사용량에 포함된다.

인터페이스 (API)

- 프로그램 작업폴더: ricehub/
- 참가자가 작성할 파일: ricehub.c 또는 ricehub.cpp 또는 ricehub.pas
- 참가자 인터페이스: ricehub.h 또는 ricehub.pas
- 견본 채점프로그램 (sample grader): grader.c 또는 grader.cpp 또는 grader.pas
- 견본 채점프로그램 입력 (sample grader input): grader.in.1, grader.in.2, ...

유의사항: 견본 채점프로그램은 다음과 같은 양식으로 입력을 읽는다:

- 1 번째 줄: R, L 과 B .
 - 2 번째 줄부터 $R+1$ 번째 줄까지: 논이 위치; 즉, $i+2$ 번째 줄에는 $X[i]$ 가 있다 ($0 \leq i < R$).
 - $R+2$ 째 줄: 예상한 해.
 - 견본 채점프로그램 입력에 대하여 예상되는 출력: grader.expect.1, grader.expect.2, ...
- 이 태스크에 대해서, 각각의 파일에는 정확히 “**Correct.**”가 포함되어 있어야 한다.