



Books

Link submit: <http://codeforces.com/problemset/problem/279/B>

Solution:

| | |
|--------|---|
| C++ | http://ideone.com/CrXp90 |
| Java | https://ideone.com/ip9i9u |
| Python | https://ideone.com/4YJjMn |

Tóm tắt đề:

Valera đến thư viện đọc sách, anh ấy lấy n quyển sách trong thư viện và anh ấy mong muốn làm thế nào để đọc được nhiều quyển sách nhất có thể. Valera phải đọc các quyển sách liên tiếp nhau.

In ra số lượng sách mà Valera có thể đọc. Lưu ý anh ấy sẽ không bắt đầu đọc quyển sách mà anh ta không đủ thời gian để đọc.

Input:

Dòng đầu tiên chứa hai số nguyên n, t ($1 \leq n \leq 10^5$; $1 \leq t \leq 10^9$) là số quyển sách và số thời gian rảnh của Valera.

Dòng thứ hai là chứa dãy n số nguyên a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^4$), mỗi số a đại diện cho thời gian để hoàn thành việc đọc quyển sách đó.

Output:

In ra một số nguyên là tổng số quyển sách Valera có thể đọc.

Ví dụ:

| | |
|----------------|---|
| 4 5 3 1 2 1 | 3 |
|----------------|---|

Giải thích ví dụ:

Có 4 quyển sách và thời gian của Valera là 5.

Valera sẽ đọc các quyển 1 2 1 để đọc được 3 quyển. Nếu chọn quyển 3 1 để đọc thì chỉ đọc được 2 quyển.

Hướng dẫn giải:

Bỏ tất cả giá trị phần tử vào mảng a. Có 2 vòng lặp chạy song song.

Vòng lặp biến i sẽ duyệt từng phần tử, các phần tử này sẽ được trừ cho thời gian t, mỗi lần trừ thì ta sẽ tăng biến đếm lên để xác định số lượng sách được đọc.

Khi biến t đã nhỏ hơn một phần tử $a[i]$ nào đó, lúc này sẽ tăng giá trị t lên bằng cách cho vòng lặp khác chạy theo j và j sẽ chạy theo sau i. Cộng giá trị $a[j]$ cho t. Lúc này ta phải giảm biến đếm lại vì lúc này đoạn chúng ta xét dịch đến một chỗ khác.

Cộng như vậy đến khi t lớn hơn hoặc bằng $a[i]$ thì ta lại đi tiếp tục tăng biến đếm lên cho và trừ $a[i]$ cho t. Khi t nhỏ hơn $a[i]$ nào đó thì ta lại lặp lại công việc là tăng j lên và cộng $a[j]$ cho t.

Độ phức tạp: $O(n)$ với n là số lượng cuốn sách.