



B. Books

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

Solution:

C/C++: <http://ideone.com/CrXp90>

Python: <https://ideone.com/4YJjMn>

Java: <https://ideone.com/ip9i9u>

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 2 số nguyên n và t ($1 \leq n \leq 10^5$; $1 \leq t \leq 10^9$) là số nguyên sách và số thời gian rảnh của Valera.

Dòng thứ 2 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 con số nguyên tổng số quyển sách Valera có thể đọc.

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

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 đọc 3 1 thì chỉ đọc được 2 quyển.

Hướng dẫn giải: Bỏ tất cả giá trị phần tử vào mảng a . Có 2 vòng lặp song song

Vòng lặp biến i sẽ chạy 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 nào đoạn chúng ta xét dịch đến một chỗ khác.

Khi cộng như vậy khi t lớn hơn $a[i]$ thì ta lại đi tiếp tục tăng biến đếm lên cho và từ $a[i]$ 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.