

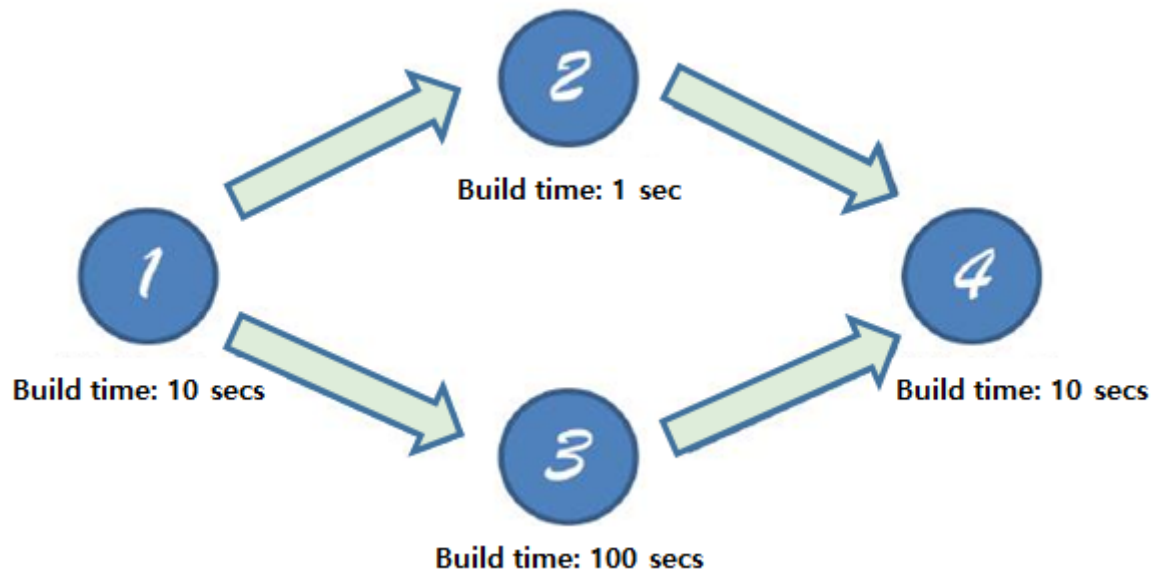
---

# ASTUCraft

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1 second  
Memory limit:         256 megabytes

The brand new game, ASTUCraft is now out for sale!

In ASTUCraft, building certain structures require other structures to already be built. Also, every structure takes a certain amount of time to build.



In the above example, you can start building structure 2 and 3 only if you finish building structure 1. (You can build structure 2 and 3 simultaneously) And to build structure 4, you need to have built both structure 2 and 3. So to build structure 4, first you need to build structure 1, which takes 10 seconds. Then, you begin building structures 2 and 3. Structure 2 takes 1 second to complete, but since you haven't finished building structure 3, you cannot build structure 4 yet. 99 seconds after that, structure 3 is built, and then you can build structure 4 for 10 seconds. Therefore, you need 120 seconds to complete building structure 4.

Given the dependency relationship between all structures, calculate the minimum time required to build a certain structure.

## Input

The first line contains  $n$ , the number of available structures, and  $k$ , the number of dependency relationships between building structures. Structures are numbered from 1 to  $n$ . ( $1 \leq n \leq 10^5, 0 \leq k \leq 2 \times 10^5$ )

The second line contains  $n$  integers,  $t_1, t_2, \dots, t_n$ , where  $t_i$  is the time required to build the structure  $i$ . ( $1 \leq t_i \leq 1000$ )

The next  $k$  lines contain two integers,  $x$  and  $y$ . This means structure  $y$  can only be built after building structure  $x$ . The last line contains  $u$ , the structure that needs to be built. ( $1 \leq x, y, w \leq n, 1 \leq u \leq 1000$ )

It is guaranteed that every structure can be built in a finite amount of time.

## Output

Print the minimum time required to build structure  $w$ .

---

## Examples

standard input	standard output
4 4 10 1 100 10 1 2 1 3 2 4 3 4 4	120
8 8 10 20 1 5 8 7 1 43 1 2 1 3 2 4 2 5 3 6 5 7 6 7 7 8 7	39