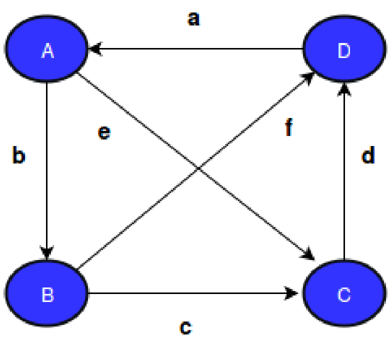
Fix the Cycles

You're given a directed weighted graph with 4 nodes (A, B, C, and D) and 6 edges, defined below:

- $D \rightarrow A$ has weight a
- $A \rightarrow B$ has weight b
- $B \rightarrow C$ has weight c
- $C \rightarrow D$ has weight d
- $A \rightarrow C$ has weight e
- $B \rightarrow D$ has weight f



The *total weight* of a simple cycle is the sum of its edge weights (e.g.: $A \rightarrow C \rightarrow D \rightarrow A$ has a total weight of $e+d+a$). If the total weight is negative, it's called a *negative cycle*.

Given edge weights a, b, c, d, e, and f, find some minimum non-negative integer (p) that, when added to *one single* edge weight in the graph, will get rid of any negative cycles.

Input Format

A single line containing 6 space-separated integers: a, b, c, d, e, and f, respectively.

Constraints

- $-20 \le a,b,c,d,e,f \le 20$

Output Format

Print the minimum value of p; if no non-negative p will eliminate the negative cycle, print -1.

Sample Input

```
2 -5 0 1 1 1
```

Sample Output

```
2
```

Explanation

Adding 2 to b (the weight of edge $A \rightarrow B$) will remove the negative cycle.