

MPT Cycle 2, Problem Set 1

Barry Hurley
Revised by Milan De Cauwer

October 31, 2015

1 Stack and Queue

Your task in this exercise is to implement both the stack and queue operations described in the lecture. Your program should read in a list of integers and insert each one into both a stack and a queue. It should then print out the contents of the stack and then the contents of the queue.

Input The first line of input contains N , the number of integers. $0 < N < 10000$. The next line of input then contains N integers.

Output Your program should output the contents of the stack in last-in first-out order, followed immediately by a new line. Then the contents of the queue in first-in first-out order.

Sample Input 1

```
9
6 9 4 1 7 8 2 3 5
```

Sample Output 1

```
5 3 2 8 7 1 4 9 6
6 9 4 1 7 8 2 3 5
```

2 Sum Free

A set of integers, \mathcal{X} , is said to be **sum-free** if the sum of any two integers in \mathcal{X} is not in \mathcal{X} . More formally:

$$\forall a, b \in \mathcal{X} : (a + b) \notin \mathcal{X}$$

Given a set of integers as input determine whether it is sum-free.

Input

The first line contains an integer N , $2 \leq N \leq 1000$, the number of integers in the set. The second line contains N space-separated positive integers.

Output

The output of your program should be 1 if the set is sum-free or 0 otherwise. The answer 1 or 0 should immediately be followed by a single newline character.

Sample Input 1

```
5
4 5 15 2 8
```

Sample Output 1

```
1
```

Sample Input 2

```
8
12 3 10 27 36 29 15 8
```

Sample Output 2

```
0
```

3 Postfix

Postfix expressions are arithmetical expressions where the operators come after anything they operate on. Postfix is important because it maintains precedence without the use of brackets. An example of a postfix expression would be:

3 5 \times

This would be evaluated as 3×5 resulting in an answer of 15.

6 4 \times 2 +

This would be evaluated as $(6 \times 4) + 2$ resulting in an answer of 26.

Input

The input will contain a postfix expression which will be no more than 50 characters in length followed by a newline and will contain one expression to be evaluated. All expressions will be correct postfix expressions. There will be one space between each number/operator. The only operators used are + (addition), - (subtraction), and * (multiplication). There is no division. All numbers are non-negative integers.

Output

Your program is to output a single integer which is the result of evaluating the postfix expression. Note that while the input numbers will not be negative, the answer may be negative.

Sample Input 1

3 5 *

Sample Output 1

15

Sample Input 2

6 4 * 2 +

Sample Output 2

26

Hint: Can the data structures from today's lecture be used?