**Problem 1**

**Output Format**

Print all  integers in  in reverse order as a single line of space-separated integers.

**Sample Input 0**

4

1 4 3 2

**Sample Output 0**

2 3 4 1

**Problem 2**

**Context**   
Given a  *2D Array*, :

1 1 1 0 0 0

0 1 0 0 0 0

1 1 1 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

We define an hourglass in  to be a subset of values with indices falling in this pattern in 's graphical representation:

a b c

d

e f g

There are  hourglasses in , and an *hourglass sum* is the sum of an hourglass' values.

**Task**   
Calculate the hourglass sum for every hourglass in , then print the *maximum* hourglass sum.

**Note:** If you have already solved the Java domain's *Java 2D Array* challenge, you may wish to skip this challenge.

**Input Format**

There are  lines of input, where each line contains  space-separated integers describing *2D Array* ; every value in  will be in the inclusive range of  to .

**Constraints**

**Output Format**

Print the largest (maximum) hourglass sum found in .

**Sample Input**

1 1 1 0 0 0

0 1 0 0 0 0

1 1 1 0 0 0

0 0 2 4 4 0

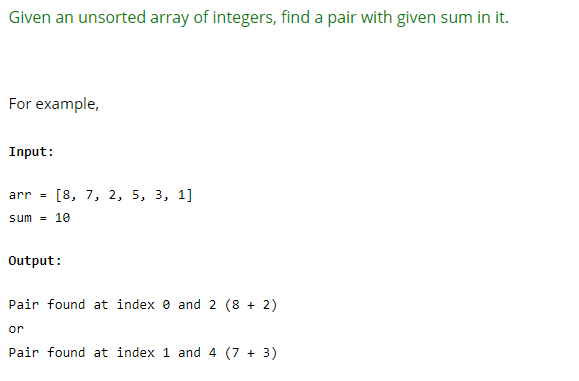
0 0 0 2 0 0

0 0 1 2 4 0

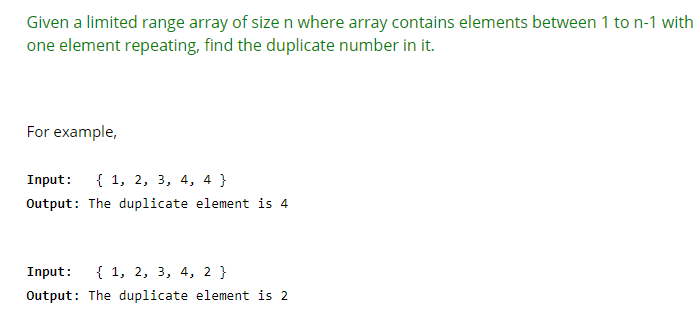
**Sample Output**

19

**Problem 3**



**Problem 4**



**Problem 5**

