Given five positive integers, find the minimum and maximum values that can be calculated by summing exactly four of the five integers. Then print the respective minimum and maximum values as a single line of two space-separated long integers.

Example

The minimum sum is and the maximum sum is . The function prints

```
16 24
```

Function Description

Complete the function with the following parameter(s):

- : an array of integers

Print

Print two space-separated integers on one line: the minimum sum and the maximum sum of of elements.No value should be returned.

Note For some languages, like C, C++, and Java, the sums may require that you use a long integer due to their size.

Input Format

A single line of five space-separated integers.

Constraints

Sample Input

```
1 2 3 4 5
```

Sample Output

```
10 14
```

Explanation

The numbers are , , , , and . Calculate the following sums using four of the five integers:

1. Sum everything except , the sum is .
2. Sum everything except , the sum is .
3. Sum everything except , the sum is .

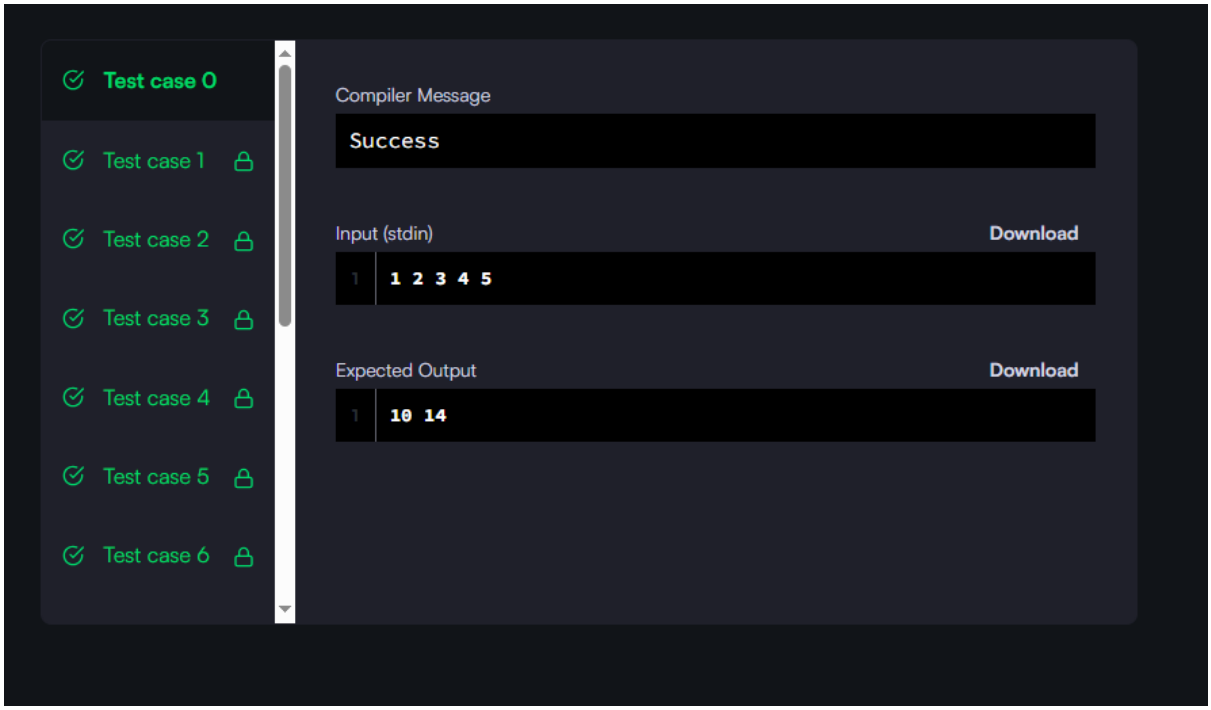
4. Sum everything except , the sum is .
5. Sum everything except , the sum is .

Hints: Beware of integer overflow! Use a 64-bit integer to store the sums.

```
import java.io.*;
import java.math.*;
import java.security.*;
import java.text.*;
import java.util.*;
import java.util.concurrent.*;
import java.util.function.*;
import java.util.regex.*;
import java.util.stream.*;
import static java.util.stream.Collectors.joining;
import static java.util.stream.Collectors.toList;

class Result {

    /*
     * Complete the 'miniMaxSum' function below.
     *
     * The function accepts INTEGER_ARRAY arr as parameter.
     */

    public static void miniMaxSum(List<Integer> arr) {
    // Write your code here
    long totalSum = 0;
     int min = Integer.MAX_VALUE;
```

```
 int max = Integer.MIN_VALUE; for (int num : arr) {
    totalSum += num;
  if (num < min) min = num;
  if (num > max) max = num;
  }
  long minSum = totalSum - max;
  // exclude largest
  long maxSum = totalSum - min;
  // exclude smallest
  System.out.println(minSum + " " + maxSum); }

  }



public class Solution {
  public static void main(String[] args) throws IOException {
    BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.in));

    List<Integer> arr = Stream.of(bufferedReader.readLine().replaceAll("\\s+$", "").split(" "))
      .map(Integer::parseInt)
      .collect(toList());

    Result.miniMaxSum(arr);

    bufferedReader.close();
  }
```

}


Test case 0
Test case 1
Test case 2
Test case 3
Test case 4
Test case 5
Test case 6
Compiler Message
Success
Input (stdin)
Download
1 1 2 3 4 5
Expected Output
Download
1 10 14