



The Maximum Possible Sum

Problem

Submissions

Leaderboard

A subsequence is any subset of an array. A subarray is a contiguous subsequence in an array.

Given an array, find the maximum possible sum among:

- All nonempty subarrays.
- All nonempty subsequences.

Print the two values as space-separated integers on one line.

Example

```
arr = [ -10, 4, 5, -4, 5, 10 ]
```

The maximum subarray sum is 20. $4 + 5 + (-4) + 5 + 10 = 20$

The maximum subsequence sum is 24. $4 + 5 + 5 + 10 = 24$

Function Description

Complete the `theMaximumPossibleSum` function in the editor below.

`theMaximumPossibleSum` has the following parameter:

- `int arr[n]`: an array of integers

Return

- `int[2]`: the maximum subarray and subsequence sums

Input Format

The first line of input contains a single integer t , the number of test cases.

The first line of each test case contains a single integer n .

The second line contains n space-separated integers $arr[i]$ where $0 \leq i < n$.

Constraints

$$1 \leq t \leq 10$$

$$1 \leq n \leq 10^5$$

$$-10^4 \leq arr[i] \leq 10^4$$

- The subarray and subsequences you consider should have at least one element.

Output Format

the subarray with the maximum sum + space + the subsequence with the maximum sum

20 24

Sample Input 0

[f](#) [t](#) [in](#)

Contest ends in 4 minutes

Submissions: 99

Max Score: 30

Difficulty: Hard



Rate This Challenge:



[More](#)

```
2
4
1 2 3 4
6
2 -1 2 3 4 -5
```

Sample Output 0

```
10 10
10 11
```

Current Buffer (saved locally, editable)  

Python 3  

```
1  #!/bin/python3
2
3  import math
4  import os
5  import random
6  import re
7  import sys
8
9  #
10 # Complete the 'theMaximumPossibleSum' function below.
11 #
12 # The function is expected to return an INTEGER_ARRAY.
13 # The function accepts INTEGER_ARRAY arr as parameter.
14 #
15
16 def theMaximumPossibleSum(arr):
17     # Write your code here
18
19 if __name__ == '__main__':
20     fptr = open(os.environ['OUTPUT_PATH'], 'w')
21
22     t = int(input().strip())
23
24     for t_itr in range(t):
25         n = int(input().strip())
26
27         arr = list(map(int, input().rstrip().split()))
28
29         result = theMaximumPossibleSum(arr)
30
31         fptr.write(' '.join(map(str, result)))
32         fptr.write('\n')
33
34 fptr.close()
35
```

Line: 1 Col: 1

 Upload Code as File ☐ Test against custom input

Run Code

Submit Code