CS2610 COA Lab exam

8 May 2021

Problem Definition

Given a choice value C and an array of n integers, A, you are expected to implement the following pseudocode.

```
if (C == 1) {
        Do sub task 1 with array A
} else if ( C == 2 ) {
        Do sub task 2 with array A
}
```

The details about the sub tasks are given below. Each sub task processes the given array and prints as output, a single integer (in the case of sub task 1) or an array of integers (in the case of sub task 2).

Sub task 1 (10 marks)

Given the array of 32-bit integers A, print the sum of all the even numbers present in the array. You can assume that the result fits in a signed 32-bit integer.

For example, if A = *[1,2,3,4]*, print $2 + 4 = 6$ as the output.

Sub task 2 (15 marks)

Given the array of 32-bit integers A, transform the array by reversing the array as described below.

$$transformed\_array = [A[n-1], A[n-2], ..., A[0]]$$

Print the transformed array as output.

An example of the transformation is shown in the figure.

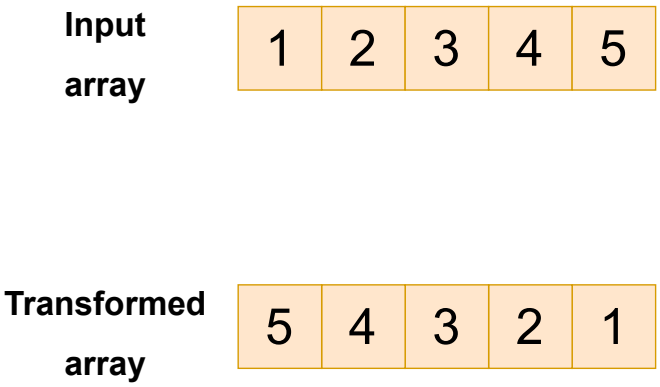

Figure 1: Sub task 2 example

Input format

- The first line of the input will contain the choice value C.
- The second line of the input will contain the number of elements in the input array n.
- The following n lines will each have a number which corresponds to the elements of the input array.

Sample input

```
2
3
77
66
55
```

Output format

- For sub task 1, the output is just a single number.
- For sub task 2, the output should be of n lines containing the elements of the output array.
- Please do not print any other text. Follow the output format strictly.

Constraints

- C will have the value 1 or 2
- n will be between 1 and 100 (both inclusive)
- The elements of the input array will have the values between 0 and 100 (both inclusive)

Testcases

Testcase 1

Input

```
1
5
13
1
10
12
5
```

Expected Output

```
22
```

Testcase 2

Input

```
1
6
8
7
10
4
2
16
```

Expected Output

```
40
```

Testcase 3

Input

```
1
20
34
37
28
16
44
36
37
43
50
22
13
28
41
10
14
27
41
27
23
37
```

Expected Output

```
282
```

Testcase 4

Input

```
2
5
13
1
```

10
12
5

Expected Output

5
12
10
1
13

Testcase 5

Input

2
6
8
7
10
4
2
16

Expected Output

16
2
4
10
7
8

Testcase 6

Input

2
20
34
37
28
16
44
36
37
43
50
22
13
28
41
10
14
27

41
27
23
37

Expected Output

37
23
27
41
27
14
10
41
28
13
22
50
43
37
36
44
16
28
37
34

Submission instructions

- The final code submission should be done on Google Form that was shared in mail.
- Submit only a single assembly file named as roll_no.asm. If your roll number is cs19b123 then submit the assembly file as cs19b123.asm
- You can submit the file only once, so check your code before submitting.