

UCF Local Programming Contest — August 30, 2014

Shopping Spree

filename: shop

(Difficulty Level: Hard)

You've won a shopping spree with a very peculiar rule. The items you are allowed to take are in consecutive order, indexed 1 through n . You may select any subset of these items subject to the following constraint:

For each index k of items chosen for the subset
at most half of the items with indexes 1 through k may be in the subset

For example, if the item with index 10 is chosen for the subset, then your selected subset can contain at most half of the items with index 1 through 10. Similarly, if the item with index 2 is chosen for the subset, then your selected subset can contain at most half of the items with index 1 through 2. Note that “half” is an integer value so half of 10 and 11 are both 5. The only exception to the constraint is that if the item with index 1 is chosen for the subset, you can select 1 item and not zero (to be fair).

The Problem:

Given a list of the dollar values of items, I_1, I_2, \dots, I_n , in the shopping spree, determine the maximum value you can obtain from the shopping spree subject to the above constraint.

The Input:

The first line of the input is a positive integer, n , indicating the number of shopping sprees that your program will have to analyze. Following this will be the descriptions of each shopping spree. Each shopping spree will be described on a single line. The first value on each of these lines will be a single positive integer, s ($s \leq 500$), representing the number of items for the shopping spree. The next s space-separated positive integers will be the values for the shopping spree items in dollars, in order. Each of these values will be less than or equal to 10^6 .

The Output:

For each shopping spree, first output the heading “Spree # d : ”, where d is the spree number, starting with 1. Then, print a single integer equal to the maximum value, in dollars, that can be obtained for that shopping spree. Follow the format illustrated in Sample Output.

Sample Input:

```
2
5
1 2 3 4 5
3
12 2 4
```

Sample Output:

```
Spree #1: 9
Spree #2: 12
```