UCF Local Contest — September 1, 2012

Hanging Nests

filename: `hanging`

The Turing Tern is a newly discovered species of bird that is known for its highly regular behavior patterns. Every member of a given flock is uniquely identifiable by the number of colored spots on its wings, i.e., no two members will have the same number of spots.

When a flock migrates to a new area, it finds a tall tree and builds a hook-like structure on the underside of the highest branch. Now each member will build a nest for itself. Every nest will hang from a hook, and will itself have up to two hooks on the underside – one on the far left, and the other on the far right.

Here are the rules used to build nests:

1) Birds always build nests in the order they arrive.
2) Every nest will hang off of exactly one of the hooks of another nest or the hook-like branch. A hook can hold at most one nest.
3) Every bird will try to hang its nest on the original hook-like branch. If that hook is being used, it will use rule (4).
4) If there is already a nest hanging on the hook that a bird is trying to use, it will compare its spots with the owner of that nest. If it has fewer, it will try to build its nest on the left hook of that nest, otherwise it will try to build it on the right hook. This procedure will repeat until it finds an empty hook.

The Problem:

The Turing Tern is endangered, and researchers have recently figured out why. When large flocks build their nests, the resulting structure is likely to become unstable. Inevitably a nest falls off the hook, taking along all the nests below it too. (While falling is rarely a problem for birds, flying is rather difficult when half a dozen nests and their occupants unexpectedly fall on your head in the middle of the night.)

We define a *hanger* of a nest as a sequence of nests connected by hooks, starting from that nest and moving one nest down at every step (never sideways) until it reaches a nest on the last level. The *hanger length* is just the number of nests in the hanger.

The *height* of a nest is defined as the length of its largest hanger. The *instability factor* of a nest is the absolute difference of the height of the nest on its left hook and the height of the nest on its right hook. (If there is no nest on a hook, the height of that 'null nest' is treated as 0.)

Your program, given a bird flock description, should output the number of spots on the bird which lives in the nest with the highest instability factor. If there are multiple such birds, output the one that arrived the earliest.

The Input:

The first input line contains a positive integer, *n*, indicating the number of flocks. Each input case takes up two lines. The first line is an integer, *f (1 ≤ f ≤ 5000)*, indicating the number of birds in the flock. The next line consists of *f* positive integers, each representing the number of spots on a bird, in the order they arrive. You may assume that no two birds have the same number of spots. Each bird has at least 1 spot and at most 5000 spots.

The Output:

For each test case, output a single line of the form "`Flock #k: p`" where *k* is the flock number, and *p* is the number of spots on the bird living in the most unbalanced nest. Leave a blank line after the output for each test case. Follow the format illustrated in Sample Output.

Sample Input:

```
3
10
10 14 12 4 8 19 18 17 9 16
5
4 5 3 2 1
15
50 67 19 42 18 66 168 1 52 57 37 64 78 22 130
```

Sample Output:

```
Flock #1: 14

Flock #2: 4

Flock #3: 66
```