ACM-ICPC Thailand Southern Programming Contest 2013

Hosted by
Department of Computer Engineering
Prince of Songkla University Hatyai Campus

10 August 2013

Contest Problems

- There are **8** problems (A-H) to solve within 3 hours 30 minutes.
- Solve as many problems as you can, in an order of your choice.
- Use C or C++ or Java to program at your convenience for any problems.
- Input and output of each program are **standard input** and **output**.

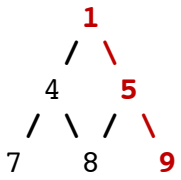
Problem A	Unlock My Safe
Problem B	Two Mysterious Alphabets from a Tree
Problem C	Max Volume
Problem D	Birthday Statistics
Problem E	Nonogram
Problem F	Jane's First Words
Problem G	Range Sum Query
Problem H	Sum of Distinct Numbers ผลรวมเลขไม่ซ้ำ

Problem B. Two Mysterious Alphabets from a Tree

Time Limit: 2s

Your task is to extract 2 alphabets from a **binary** tree which is composed of unsigned integers respecting the following rules. Let n be the height of a tree. At the level k ($1<=k<=n$), the tree contains k of nodes and each node has 2 children nodes (except the leaf nodes at the level n which have no children). See the example below to understand the tree formation. Some nodes may have 2 parent nodes.

Example:



You need to walk in a tree on the path that has a maximum summation (e.g., 1 + 5 + 9 = 15). Numbers in each summation cannot cross into different links (e.g., 5+7 is illegal). Then, your intermediate task is to calculate *2 numbers* for alphabet extraction. *The first number* is calculated from $\sum_{i=1}^{n} i^2$ where i is a number along the maximum summation path and n is the height of a tree. *The second number* is a summation of the maximum path ($\sum_{i=1}^{n} i$). Regarding to the example above, the first number = 1 + 25 + 81 = 107 and the second number = 1 + 5 + 9 = 15.

Finally, these two numbers are transformed into two lower case alphabets from 'a' to 'z' respectively, where 'a' is used for 0 and 'z' is used for 25. Since there are only 26 alphabets, a number greater than 25 will reuse the same set of alphabets. For example, 107 = 'd' and 15 = 'p' (that is, the first alphabet 'a' = 0, or 26, or 52 etc).

Write a program to find the 2 mysterious alphabets from a given tree.

Input

The first line of input contains the height (n) of a tree ($0<n<100$). The second line contains unsigned integer numbers (i) in each level of a tree ($0<i<100$), consecutively. *Assume that there is only one maximum path in a tree*.

Output

The first line contains two integer calculated from the rules above, and the second line contains 2 decoded alphabets.

Sample Input	Sample Output
3 1 4 5 7 8 9	107 15 dp

Sample Input	Sample Output
4 1 5 2 5 1 9 3 4 20 1	486 32 sg

Sample Input	Sample Output
5 2 4 9 1 3 1 1 1 1 2 12 5 4 3 2	166 20 ku

Sample Input	Sample Output
6 9 8 8 7 7 7 9 1 1 3 8 2 10 5 1 2 3 2 1 9 81	6765 109 ff