CCC 计算机竞赛预习材料 – Sample Questions

Write the output for following questions

1) A planetary "rover" is traveling on a perfectly level surface (no obstacles) on Mars. The rover is in radio communication with the "lander" it arrived in. The lander accumulates and relays commands, which it receives from Earth, to the rover. The rover makes several excursions. Each excursion begins with the rover at the lander facing in a known direction. At the end of each excursion the lander must compute and transmit a sequence of instructions to return the rover to the lander. The rover responds to a sequence of commands sent from the lander. Each command tells the rover to move ahead some multiple of an exact unit of distance, or to turn left or right exactly 90 degrees. The "move ahead" command is encoded using two consecutive lines in the input file. The first line contains the integer 1, and the second line contains a non-negative integer n, the distance to move forward. The "turn right" command is encoded using a single input line containing the integer 2. The "turn left" command is encoded using a single input line containing the integer 3.

   The input file begins with a line containing a positive integer, n, the number of excursions for the rover. The commands for the excursions occupy subsequent lines of the input file. Each excursion consists of a number of commands followed by a line containing 0. There are no errors or blank lines in the input. The rover travels less than 10 000 units of distance on each excursion.

   For each excursion, the output file should contain a line:

   Distance is k

   where k is the distance in units that the rover must travel to return to the lander. The following lines should contain the shortest sequence of commands to return the rover to the lander. A blank line should separate the lines of output for different excursions.

   Sample input

   3

   2

   3

   3

   0

   3

1

10

2

1

5

0

1

5

2

1

5

3

3

1

1

0

What should be the output?

Distance is 0

Distance is 15

2

1

10

2

1

5

Distance is 9

1

4

3

1

5

2) A series of streams run down the side of a mountain. The mountainside is very rocky so the streams split and rejoin many times. At the foot of the mountain, several streams emerge as rivers. Your job is to compute how much water flows in each river. At any given elevation there are n streams, labelled 1 to n from left-to-right. As we proceed down the mountainside, one of the streams may split into a left fork and a right fork, increasing the total number of streams by 1, or two streams may rejoin, reducing the total number of streams by 1. After a split or a rejoining occurs, the streams are renumbered consecutively from left-to-right. There is always at least one stream and there are never more than 100 streams. The first line of input contains n, the initial number of streams at some high altitude. The next n lines give the flow in each of the streams from left-to-right. Proceeding down the mountainside, several split or rejoin locations are encountered. For each split location, there will be three lines of input; a line containing 99 (to indicate a split) a line containing the number of the stream that is split a line containing a number between 0 and 100, the percentage of flow from the split stream that flows to the left fork. (The rest flows to the right fork). For each join location, there will be two lines of input; a line containing 88 (to indicate a join) a line containing the number of the stream that is rejoined with the stream to its right The flow from both joined streams is combined. After the last split or join location will be: a single line containing 77 (to indicate end of input) Your job is to determine how many streams emerge at the foot of the mountain and what the flow is in each. Your output is a sequence of real numbers, rounded to the nearest integer, giving the flow in rivers 1 through n.

Sample Input

3

10

20

30

99

1

50

88

3

88

2

77

What should be the output?

5 55

3) In a number of sports, a championship may be determined by a double knockout competition. A team is eliminated on its second loss, so the winner is the last remaining team with one or fewer losses. The competition is played is a series of rounds: in each round, teams that have not been eliminated are paired subject to the constraint that an undefeated team never plays a team with one loss. As many teams as possible are paired in each round. After a number of rounds only two teams remain. These teams play in a round by themselves, although one is undefeated and the other is not. If neither is eliminated, they play again in a final round. For our analysis we assume that this extra round is always necessary.

The first line of input contains one positive integer n which is the number of test cases which follow it. The next n lines each contain one positive integer t, t < 32768, which is the number of teams in the competition for that test case. For each case there should be an initial line which has the form: Round 0: 2 undefeated, 0 one-loss, 0 eliminated This is followed by a similar line

for each round of the competition, followed by a single line saying how my rounds were played. The output for different test cases is to be separated by a single blank line.

Sample Input

1

2

What should be the output?

Round 0: 2 undefeated, 0 one-loss, 0 eliminated

Round 1: 1 undefeated, 1 one-loss, 0 eliminated

Round 2: 0 undefeated, 2 one-loss, 0 eliminated

Round 3: 0 undefeated, 1 one-loss, 1 eliminated

There are 3 rounds.