

Problem 1: A Game

Consider the following two-player game. The game board consists of a sequence of positive integers. The two players move alternately. When a player moves, he selects a number from either the left or the right end of the sequence. The selected number is deleted from the board. The game is over when all numbers have been selected. The first player wins if the sum of the numbers he has selected is at least as much as selected by the second player. The second player plays his best.

The first player starts the game.

If the board initially contains an even number of elements, then the first player has a winning strategy. You are to write a program that implements the strategy of the first player to win the game. The second player's response is provided by a given computer program. The two players communicate with three procedures of the [module Play](#) that is made available to you. These procedures are `StartGame`, `MyMove` and `YourMove`. The first player should initiate the game by executing the parameterless procedure `StartGame`. If the first player selects a number from the left end, he executes the procedure `MyMove('L')`. Similarly, executing the instruction `MyMove('R')` sends a message to the second player indicating that the first player has selected a number from the right end. The second player, i.e. the machine moves immediately, and the first player can learn this move by executing the instruction `YourMove(C)`, where `C` is a character type variable (in C/C++ you write this as `YourMove(&C)`). The value of `C` is 'L' or 'R' depending on whether the selection has been made either from the left or the right end.

Input Data

The first line of file `INPUT.TXT` contains the size `N` of the initial board. `N` is even and $2 \leq N \leq 100$. The remaining `N` lines contain one number in each line, the contents of the initial board in left to right order. Each number is at most 200.

Output Data

When the game is over, your program should write the final result of the game to the file `OUTPUT.TXT`. The file contains two numbers in the first line. The first number is the sum of the numbers selected by the first player and the second number is the sum of the numbers selected by the second player. Your program must play a game and the output must correspond to the game played.

Example Input and Output

Figure 1 gives an input file containing an initial board description and a possible output file.

INPUT.TXT

6

4

7

2

9

5

2

OUTPUT.TXT

15 14