Assignment 2

Introduction to programming in C

Question 1

You are given a non-negative sequence of numbers, ending with a -1. You can assume that there are at least two numbers before the ending -1.

You have to output the second largest element of the sequence. If there is no second largest element in the sequence then output 0.

Note : -1 is not a part of input. It only signifies that input has ended.

Solution

```c
#include<stdio.h>

int main() {
  int largest = 0, second = 0;
  int n = 0;

  while(1) {
    scanf("%d", & n);
    if (n == -1) {
      printf("%d", second);
      break;
    }
    if (n > largest) {
      second = largest;
      largest = n;
    }else if (n > second && n != largest) {
      second = n;
    }
  }
  return 0;
}
```

Question 2

You are given a non decreasing sorted sequence of non negative integers, ending with -1. That is if the sequence is $a_1, a_2, \ldots, a_n, -1$ then $a_i \leq a_{i+1}$ for all i from 1 to n-1. You can assume that are at least two numbers before the ending -1. You have to output the number of distinct elements in the sorted sequence.

Solution

```c
#include <stdio.h>

int main() {
  int curr = 0; /* current value being read */
  int prev = 0; /* previous value read */
  int num_distinct = 0; /* number of distinct values read */

  scanf("%d", &curr);
  num_distinct = 1;

  while (curr != -1) {
    prev = curr;
    scanf("%d", &curr);
    if (prev != curr && curr != -1) {
      num_distinct = num_distinct + 1;
    }
  }
  printf("%d", num_distinct);
  return 0;
}
```

Question 3

In this assignment, you will be given an $N \times N$ matrix, with $N > 1$.

You have to determine whether the matrix is an upper triangular matrix.

A matrix is upper triangular if every entry below the diagonal is 0. The following is an example of an upper triangular matrix:

$$\begin{matrix} 1 & 1 & 1 & 1 \\ 0 & 4 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{matrix}$$

Note: The diagonal itself, and the entries above the diagonal can be zeroes or non-zero integers.

Input

First, you will be given N, which is the size of the matrix. Then you will be given N rows of integers, where each row consists of N integers separated by spaces.

Output

If the input matrix is upper triangular, then print 1. Otherwise, print 0.

Solution

```
#include<stdio.h>

int main() {
  int i, j, c, flag, n;
  flag = 1;

  scanf("%d", & n);

  for (i = 1; i < n + 1; i++) {
      for (j = 1; j < n + 1; j++){
          scanf("%d", &c);

          if(i > j && c!=0){
              flag = 0;
          }
      }
  }
  printf("%d",flag);
  return 0;
}
```