Assignment 2 Solutions

Q. 1 Level: Medium

In this assignment, you will be given an NxN matrix. You have to determine whether the matrix is a triangular matrix.

The diagonal of the matrix M of size NxN is the set of entries M(0,0), M(1,1), M(2,2), ..., M(N,N).

A matrix is upper triangular if every entry below the diagonal is 0. For example, 
\[
\begin{pmatrix}
1 & 1 & 1 \\
0 & 0 & 1 \\
0 & 0 & 2
\end{pmatrix}
\]
is an upper triangular matrix. (The diagonal itself, and the entries above and below the diagonals can be zeroes or non-zero integers.)

A matrix is lower triangular if every entry above the diagonal is 0. For example, 
\[
\begin{pmatrix}
2 & 0 & 0 \\
3 & 1 & 0 \\
4 & 2 & 2
\end{pmatrix}
\]
is a lower triangular matrix.

A matrix is triangular if it is either upper triangular or lower triangular or both.

You may not use arrays for this program.

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 triangular, then print yes. Otherwise, print no.

Sol.

```c
#include<stdio.h>
int main()
{
    int i,x,p,q,n,counter,z;
    p=0;
    q=0;
    scanf("%d",&n);
    for(i=1;i<n+1;i++)
    {
        counter =1;
        for(;counter<i;)
        {

```
```c
scanf("%d", &z);
counter++;
if(z == 0)
else
    p = 1;
}

if(counter == i)
{
    scanf("%d", &x);
counter++;
}
for(; counter < n + 1;)
{
    scanf("%d", &z);
counter++;
    if(z == 0)
else
    q = 1;
}

if(p == 1 && q == 1)
    printf("yes");
else
    printf("no");
return 0;
```
Q.2 You are given a sorted (either in the increasing or in the decreasing order) sequence of numbers, ending with a -1. You can assume that are at least two numbers before the ending -1.

Let us call the sequence $x_0 \ x_1 \ ... \ x_{n-1}$.

You have to output the number of distinct elements in the sorted sequence.

Kindly do not use arrays in the code.

Sol.

```c
#include <stdio.h>
int main()
{
    int curr=0; /* current value being read */
    int prev; /* previous value read */
    int num_distinct=0; /* number of distinct values read */
    
    prev = curr;
    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\n", num_distinct);
    return 0;
}
```
Q.3 You are given a sequence of non-negative integers terminated by -1. You have to output 1 if there are at least 2 distinct elements in the sequence and 0 if the sequence consists of only 1 integer. Note that -1 is not part of the sequence. The sequence is not necessarily sorted.

Note: Don't use arrays to this question.

Sol.

```c
#include<stdio.h>

int main()
{
    int a,b;
    scanf("%d",&a);
    for(;;)
    {
        scanf("%d",&b);
        if(b==-1)
        {
            printf("0");
            break;
        }
        if(a!=b)
        {
            printf("1");
            break;
        }
    }
    return 0;
}
```