Name:……………………   Q11

CS 1713 Intro to Programming II

Suppose we have the following structure (record) declaration.

```
typedef struct {
   int x;
   int y;
} myDataT;
```

Write a program that dynamically allocates a triangular-like 2D array of the above structure with the given number of rows denoted by N such that the first row (row 0) will have one record, second row (row 1) will have two records, and so on. The last row (row N-1) will have N records. For example, when N is 5, conceptually the 2D array will look like

x=?
y=?				

x=?
y=?
	x=?
	y=?			

x=?
y=?
	x=?
	y=?
		x=?
		y=?		

x=?
y=?
	x=?
	y=?
		x=?
		y=?
			x=?
			y=?	

x=?
y=?
	x=?
	y=?
		x=?
		y=?
			x=?
			y=?
				x=?
				y=?

After allocating then memory, your program should initialize x and y fields in each cell by setting them to corresponding cells' row numbers and column numbers, respectively. So after the initialization, the above array will look like:

x=0
y=0				

x=1
y=0
	x=1
	y=1			

x=2
y=0
	x=2
	y=1
		x=2
		y=2		

x=3
y=0
	x=3
	y=1
		x=3
		y=2
			x=3
			y=3	

x=4
y=0
	x=4
	y=1
		x=4
		y=2
			x=4
			y=3
				x=4
				y=4

Finally, your program should free up (release) the allocated memory.

Complete the following C program which (i) dynamically allocates a triangular-like 2D array, as described in previous page (ii) initialize it, and (iii) free up the allocated memory.

```
#include <stdio.h>
void main(void)
{
  int N, i, j;
  myDataT **a;
  printf(“Enter Number of Rows : “); scanf(“%d”, &N);
```

```
typedef struct {
     int x;
     int y;
} myDataT;
```