CS 22203: Design and Analysis of Algorithms, Jan-May, 2025
Practical Assignment 4

1. Consider the following function Guess(x,n) (assume n >= 1)

```
int Guess(int x, int n){

        if (n==1)

                return x;

        if (n is even)

                return (Guess (x+x, n/2));

        else

                return (x + Guess (x+x, n/2));

}
```

a) Implement the above function. Try out for different values of x and n. And guess what it does.

b) Modify the algorithm so that it displays strings that represent nodes in the recursion tree (as taught in the class). Run the algorithm when the values of (x,n) are (3,7), (5,20), (6,9) respectively. The output for (3,7) will look like this:

Guess(3,7)

Guess (6,3)

Guess(12,1)

c) Try drawing the recursion tree using any available Python module.

2. The recursion tree of Q. no.1 is a tree in which an intermediate node has only 1 child (the tree looks like a string of beads). However, *Mergesort* algorithm will generate a more general kind of recursion tree. Implement the *Mergesort* and the *Merge* procedures and generate the strings denoting nodes (calls to Mergesort and Merge) in the recursion tree. Get a random set of *n* elements and try out. Draw the recursion tree using any available Python module.