Prac: Recursion

1. Purely by studying the following code listing, can you predict *exactly* what the program would output when executed?

```
class Testcheers
{
    public static void Main( string[ ] args )
    {
        int x = 3;
        cheers(  x  );
    }

    public static void cheers( int n )
    {
        if( n == 0 )
        {
            Console.WriteLine( “Hurray” );
        }
        else
        {
            Console.WriteLine( “Hip  ” );
            cheers( n – 1 );
        }
    }
}
```

   Test your prediction by copying the code, compiling and running it.

2. Write a program that uses a recursive method to print numbers from *n* (input by user) to 0. Remember a loop must not be used. The ‘Testcheers’ program may help indicate a solution.

3. The factorial of a number is that number multiplied by itself less 1 with this action repeated until the multiplier itself is equal to 1. For example, the factorial of 5 (symbolized as 5!) would be 5x4x3x2x1 = 120. An earlier practical required the development of a program to solve a factorial input by the user using using a loop structure. Now write a program that uses recursion to solve this problem (ie it should *not* use any kind of loop structure). If you find this too difficult, try a web search to find an algorithmic pseudocode recursive solution that you can use to implement in C#.

4. To visualize how recursion works for some cases of sorting, view the following URL’s.

   http://www.sorting-algorithms.com/
   https://www.bluffton.edu/~nesterd/java/SortingDemo.html