Homework 1 Due:30/10/2022

Q1   [60p] Write a function named `void generate(int n)` that produces the pattern shown in Figure 1 and Figure 2 for any input $2 < n < 30$ that is going to be read using `scanf`. Use `printf("%3d",...)` when printing the numbers. Hard coding is not allowed, you have to use only `for` loops and `printf("%3d",...)`.
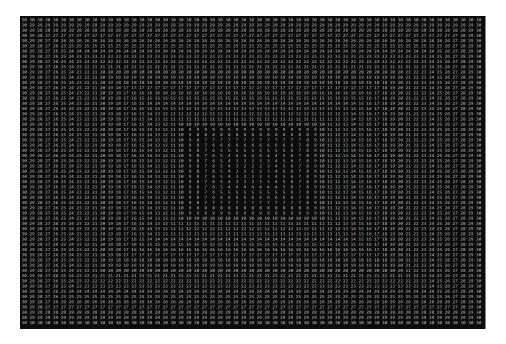

Figure 1: Results of Q1

Q2   [40p] Implement the Newton's method using recursion.   The function prototypes `double pol(double x);` and `double dpol(double x);` are given. The function `double iterate(double);` needs to implement the recursive algorithm. The Newton's method is given below. Let `pol(x)` be a differentiable function. The solution to $p(x) = 0$ can be calculated using an initial guess (zero in the implementation) and the following relation between iterations,

$$x_{n+1} = x_n - \frac{pol(x_n)}{dpol(x_n)}$$

- Submit a single `*.c` file to NINOVA. Other file types will not be accepted nor graded.

- The given main function is not going to be submitted, only the necessary implementation needs to be submitted.

- Your submission will be compiled with a tester `main.c` file. Your code needs to compile without error, or your grade will be zero.

- Each functionality will be tested and added to your grade.

- Late submissions will be deduced $10p$ for each day late.

- Cheating is not allowed, once cheating is detected all involved submissions will be graded zero.

Figure 2: Results of Q1

```c
//*****************************************
//** DO NOT SUBMIT THIS FILE
//*****************************************
#include <stdio.h>
#include <math.h>

/*   Q1 */
void generate(int);
/*   Q2 */
double pol(double); //implementation may go here
double dpol(double);//implementation may go here
double iterate(double);

int main()
{
    /*   Q1 */
    // some code to test the function(s)
    /*   Q2 */
    // some code to test the function(s)

    return 0;
}
```
                              deliverables/main.c

```c
//*****************************************
//** SUBMIT ONLY THIS FILE
//** DO NOT CHANGE ANYTHING
//** ONLY ADD THE IMPLEMENTATION
//*****************************************
#include <stdio.h>
#include <math.h>

//***************** Q1 *****************
void generate(int n)
{
    //implementation goes here
}

//***************** Q2 *****************
double pol(double); //do not implement here
double dpol(double);//do not implement here
double iterate(double a)
{
    //implementation goes here
    return a;
}
```

deliverables/student.c