

## Teoria da Computação – Lista de Exercícios 1:

Represente os programas em C abaixo na forma de **programas monolíticos, iterativos e recursivos**, conforme definições apresentadas em sala de aula.

Em seguida, codifique os programas desenvolvidos usando as macros dos arquivos `mono.c`, `iter.c` e `recur.c`. Por fim, defina máquina (operações e testes) necessários para a execução dos programas, seguindo modelo dos arquivos `um_reg.cpp` e `dois_reg.cpp`.

### I – Fatorial

```
#include <stdio.h>
int fat (int n);
int main (void){
    int n, r;
    scanf("%d", &n);
    r = fat(n);
    printf("Fatorial = %d\n", r);
    return 0;
}

int fat (int n){
    int i;
    int f = 1;
    for (i = 1; i <= n; i++)
        f *= i;
    return f;
}
```

### II – Raiz quadrada

```
#include <stdio.h>
// defining function prototype
float square_root(float);
//Now the main program starts
int main()
{
    float n;
    printf("Enter the number: ");
    scanf("%f",&n);
    printf("\n");
    printf("square root is: %f",square_root(n));
    return 0;
}

//using the babylonian method
float square_root(float n)
{
    float x = n;
    float y = 1;
    float a = 0.000001; // for cloase approximation
    while((x-y) > a)
    {
        x = (x+y)/2;
        y = n/x;
    }
    return x;
}
```