

UNIVERSIDADE DO ESTADO DE SANTA CATARINA
ANA CAROLINA VEDDY ALVES
COMPLEXIDADE DE ALGORITMOS
LISTA 1

1) *Implemente uma função que calcule o n-ésimo termo da sequência de Fibonacci com complexidade de tempo linear. Construa um gráfico comparando a execução da versão linear com a versão recursiva definida abaixo:*

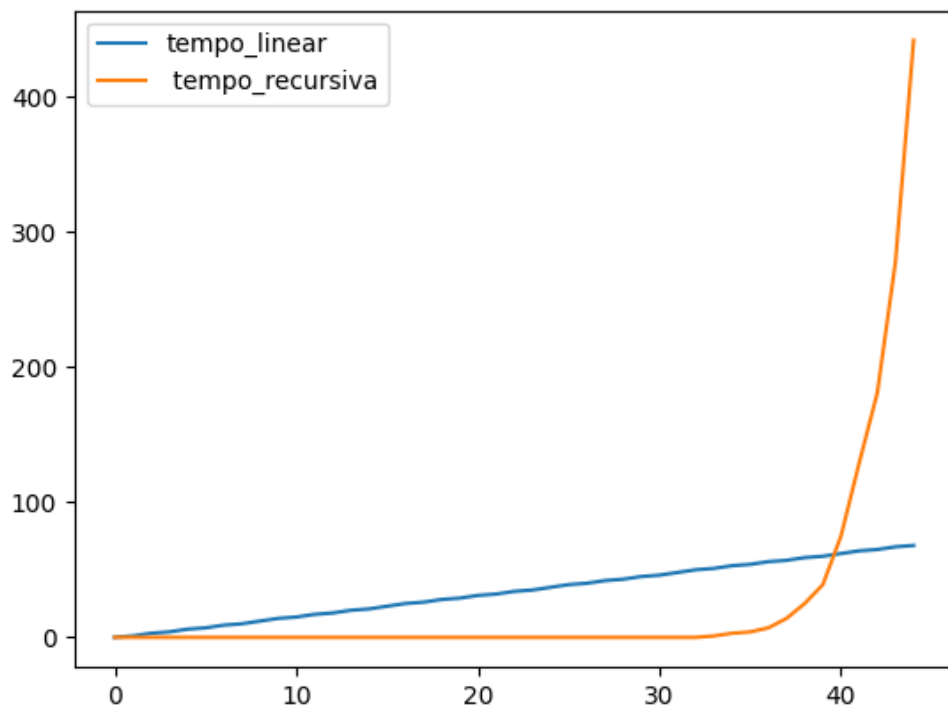
```
unsigned int fib (unsigned int n) { if (n < 2) return n; return fib (n-2) + fib (n-1); }
```

Resposta:

Função utilizada:

```
unsigned int fib_linear(unsigned int n){  
    int a=0; int b=1; int n_fib = 0;  
    for(int i=0;i<n;i++){  
        n_fib = a+b; a=b; b=n_fib; }  
    return n_fib; }
```

Comparação de tempos dos algoritmos sobre n-ésimo número da sequência de Fibonacci:



Os códigos implementados estão na pasta *exercicio_1* desta entrega. Para executá-los basta executar num terminal Linux o comando:

```
gcc exercicio_1_fibonacci.c -o exercicio_1_fibonacci && ./exercicio_1_fibonacci
```

E em seguida:

```
python3 graphics.py
```

É necessário ter as seguintes bibliotecas instaladas:

- Pandas (para instalar basta executar o comando *pip install pandas*)
- Matplotlib (para instalar basta executar o comando *pip install matplotlib*)

Os gráficos estarão no arquivo chamado *tempos_comparacao.png*.