

# EXERCÍCIOS SOBRE THREADS

(respostas)

1- Thread (linha de execução) é uma sequência de instruções que faz parte de um processo principal. Um software é organizado em processos. Cada processo é dividido em threads, que formam tarefas independentes, mas relacionadas entre si. CPUs podem realizar multithreading simultâneo (SMT) para ter mais desempenho.

2- Programas multiencadeados podem melhorar o desempenho em comparação com os programas paralelos tradicionais que utilizam vários processos. Além disso, o desempenho melhorado pode ser obtido em sistemas multiprocessador usando threads.

3- Kernel Threads: baratos, não são associado com processos de utilizador, e têm a sua própria pilha. Lightweight Process: kernel supported user-thread. User Threads: implementados por bibliotecas (C-threads ou pthreads).

## EXERCÍCIO PRÁTICO

1-

```
import time
import threading

lock = threading.Lock()

def contagemDeNumeros():
    for i in range(1,11):
        print(i, end = " ")
        time.sleep(1.5)

def contagemLetras():
    letras = ["A", "B", "C", "D", "E", "F", "G", "H", "I", "J"]
    for i in letras:
        print(i)
        time.sleep(1.5)

thread1 = threading.Thread(target = contagemDeNumeros)
thread2 = threading.Thread(target = contagemLetras)
```

```
thread1.start()
thread2.start()

thread1.join()
thread2.join()

print("Programa finalizado!")
```

2-

```
import time
import threading

lock = threading.Lock()

def contagemDeNumeros():
    for i in range(1,11):
        print(i)
        time.sleep(1.5)

def contagemLetras():
    letras = ["A", "B", "C", "D", "E", "F", "G", "H", "I", "J"]
    for i in letras:
        print(i)
        time.sleep(1.5)

def contagemDeNumerosInversa():
    for i in range(10,0,-1):
        print(i)
        time.sleep(1.5)

def contagemLetrasInversa():
    letras = ["A", "B", "C", "D", "E", "F", "G", "H", "I", "J"]
    for i in letras[len(letras):0:-1]:
        print(i)
        time.sleep(1.5)

thread1 = threading.Thread(target = contagemDeNumeros)
thread2 = threading.Thread(target = contagemLetras)
thread3 = threading.Thread(target = contagemDeNumerosInversa)
thread4 =
```

```
threading.Thread(target = contagemLetrasInversa)

thread1.start()
thread2.start()
thread3.start()
thread4.start()

thread1.join()
thread2.join()
thread3.join()
thread4.join()

print("Programa finalizado!")
```