

LINGUAGEM DE PROGRAMAÇÃO II

Centro de Informática – Universidade Federal da Paraíba – Campus I

Professor: Carlos Eduardo Batista - Semestre: 2020.1

Entrega por e-mail: 23h59 de 05/12/2020.

O título do e-mail deve conter: “[LPII-20201-E003] NOME DO ALUNO – MATRICULA”. Arquivo de entrega deve anexar todos os códigos fonte em C/C++ dentro de um diretório nomeado “MATRICULA_ALUNO-LPII-20201-E003” o qual deve ser comprimido em um arquivo ZIP (“MATRICULA_ALUNO-LPII-20201-E003”). O arquivo ZIP deve conter obrigatoriamente um arquivo de texto chamado README.txt contemplando todas as instruções de compilação e execução, além de qualquer observação que se fizer necessária para correção.

TRABALHO PRÁTICO: *BARREIRAS*

Faça um programa em C/C++ onde 6 *threads* (além da principal) gerarão uma letra aleatória (caractere) cada. Estas letras deverão ser combinadas em uma palavra (uma *string*) na thread principal (função *main*), que deverá exibi-la e também determinar se esta é um palíndromo ou não. Palíndromos são frases ou palavras que podem ser lidas, indiferentemente, da esquerda para a direita, conforme o sentido habitual da leitura, ou da direita para a esquerda, sem que haja mudança da sua significação (exemplos de palíndromos com 6 letras: “AAAAAA”, “ABAABA”, “ABCCBA”, “MUSSUM”). Repita este processo 10 vezes e use barreiras para sincronizar as *threads*.

A implementação deverá ser feita em C/C++ utilizando a biblioteca Pthreads ou as threads STD (C++11). A implementação deverá atender às quatro propriedades de uma solução para o problema da seção crítica: exclusão mútua, ausência de *deadlock*, ausência de atraso desnecessário e entrada eventual. A saída do seu programa deve ser bem planejada, de forma a mostrar o que está acontecendo em cada momento relevante.