

Instruções para entrega: Enviar os seguintes arquivos para *mlc2@cin.ufpe.br* : (1) arquivos .hs e .java, e (2) um arquivo no formato pdf com **todas** as respostas, incluindo aquelas que não envolvem código apenas. Escrever o nome na primeira linha dos arquivos. O assunto do email deve estar no formato: Nome completo_2ee_plc_2019_1.

- (1.5) 1. (Java) Considere a classe abaixo

```
public class C extends Thread{
    private Object a;
    private Object b;

    public C(Object x, Object y){
        a = x;
        b = y;
    }

    public void run(){
        synchronized(b){
            synchronized(a){ ... } } }
}
```

Considere também o seguinte código, em que a1 e b1 são do tipo Object.

```
C t1 = new C(a1, b1);
C t2 = new C(a1, b1);
t1.start();
t2.start();
```

Há possibilidade de haver *deadlock* ao executarmos o código acima? Justifique. Se sim, explique em detalhes como pode acontecer, descrevendo uma execução intercalada problemática e os *locks* utilizados.

- (3.0) 2. (Java) Implemente em Java um produtor-consumidor de valores inteiros.
- (3.0) 3. (Haskell) Na preparação de sanduíches em uma lanchonete, uma pessoa fornece os ingredientes (pão, carne e tomate); duas outras são responsáveis por preparar os sanduíches. Porém, a lanchonete dispõe de apenas uma faca para ser utilizada na preparação. Considere que os recipientes de ingredientes são continuamente reabastecidos na capacidade máxima de porções (30 para cada ingrediente). Desenvolva uma solução em Haskell que modele o funcionamento desta lanchonete.
- (2.5) 4. (Lua) Escreva uma função que recebe os coeficientes de um polinômio e retorna uma função que, quando chamada com um valor para x, retorna o valor do polinômio para aquele x. Por exemplo,

```
f = newpoly({3,0,1})
print(f(0))    - - > 1
print(f(5))    - - > 76
print(f(10))   - - > 301
```

$$5x^2 + 5.0 + 1$$
$$25 + 1$$
$$3.100 + 1$$