

Nome: \_\_\_\_\_ Matrícula: \_\_\_\_\_

- 1) a) Qual(is) a(s) vantagem(ns) da utilização da técnica de semáforos sobre o uso de Monitores?
- b) Qual(is) a(s) vantagem(ns) da utilização da técnica de Monitores sobre o uso da técnica de espera ocupada (usando apenas variáveis compartilhadas) ?
- c) Qual(is) a(s) vantagem(ns) da utilização da técnica de Monitores sobre o uso da técnica de Semáforos ?

2) Considere o código mostrado a seguir.

a, b, c, d, e : semaphore initial 0;

P1: ... P(a); P(a) “código de P1” V(b); V(b);	P2: ... P(b); “código de P2” V(c);	P3: ... “código de P3” V(a);	P4: ... “código de P4” V(a);
P5: ... P(b); “código de P5” V(d);	P6: ... P(c); “código de P6” V(e);	P7: ... P(d); “código de P7” V(e);	P8: ... P(e); P(e) “código de P8”

- a) Faça um desenho do grafo de precedência das tarefas P1 a P8 definidas pelo código acima.
- b) Suponha que os processos sejam executados de forma cíclica. Ou seja, após o término do último processo do grafo que você desenhou, a mesma sequência de processos seria executada novamente e de forma repetitiva (loop infinito). Quantos novos semáforos seriam necessários para tornar esta modificação possível. Qual o valor inicial de cada novo semáforo?
- c) Mostre a modificação no código dos processos P1 a P8 que seria necessária implementar para a execução na forma definida no item “b”.

3) Considere o problema da cervejada, descrito a seguir.

“Jovens em uma cervejada passam o tempo todo bebendo e conversando. Quando um jovem sente sede ele vai até o balcão de um quiosque e entra em uma fila indiana para pegar uma (e apenas 1) lata de bebida. Existem dois tipos de jovens: os que tomam cerveja ou refrigerante e os que tomam apenas refrigerante. Os que tomam cerveja, só tomam refrigerante quando não encontram cerveja. Quando um jovem chega ao balcão e não tem mais a bebida que ele pode pegar, ele chama o atendente do quiosque, que pega um lote da bebida preferida do jovem (48 latas de cerveja ou 24 latas de refrigerante) e coloca as latinhas no balcão. No início da cervejada o atendente coloca o lote inicial de 48 latas de cerveja e 24 latas de refrigerante no balcão.”

Implemente uma solução para o problema da cervejada usando Monitores.

```
process Jovem_Cerveja(i : 1 to 10)
  k : boolean;          /* quando k é true o jovem conseguiu pegar um a cerveja */
  loop
    ...
    Balcao.pega_cerveja(k);      /* solicita uma cerveja, mas pode receber cerveja ou refrigerante */
    if k then "bebendo cerveja"; /* bebendo cerveja junto com outros jovens */
      else "bebendo refrigerante"; /* bebendo refrigerante junto com outros jovens */
    ...
  endloop

process Jovem_Refri(j : 1 to 5)
  loop
    ...
    Balcao.pega_refri();      /* solicita um rfrigerante */
    "bebendo";                /* bebendo refrigerante junto com outros jovens */
    ...
  endloop

process Atendente;
  loop
    ...
    Balcao.repoe_bebida;      /* o tipo de bebida a ser reposta será definida por variável
                               interna do Monitor */
    ...
  endloop;
```

4) Em um parque de diversões um dos brinquedos mais populares é a montanha russa. Os clientes (passageiros) que conseguem dar uma volta, sempre tentam dar mais uma voltinha. O carro da montanha russa tem capacidade para  $C$  passageiros e uma volta só pode ser iniciada quando o carro estiver com os  $C$  passageiros. Existem  $N$  passageiros tentando dar voltas, com  $N > C$ . Um passageiro entra no carro e espera que o carro inicie a volta (os  $C$  passageiros que conseguiram entrar no carro iniciarão a volta ao "mesmo tempo" - concorrentemente). Ao terminar a volta, todos os passageiros tem que sair do carro e para dar outra volta eles têm que voltar para o final da fila de entrada da montanha russa. Novos passageiros só podem entrar no carro, após todos os passageiros que deram a última volta saírem do carro.

Implemente uma solução usando Regiões Críticas Condicionais para resolver a sincronização entre os processos passageiros e o processo carro.