Roteiro de Estudos 7
Sincronização - Semáforos

1. **LEITURA**
    - Silberschatz, P. Baer Galvin, e G. Gagne, , "**Fundamentos de Sistemas Operacionais**", 8a. Edição, Editora LTC, 2010.
        - Seções 7.4, 7.5 e início da 7.6

2. **RESUMÃO SOBRE SEMÁFOROS**
    - Slides com uma compilação do conteúdo

3. **VÍDEO**
    - Semaphores (Xoviabcs) (~9'00'')
        - No link estamos pulando a explicação sobre as primitivas sleep/wakeup (tive que reduzir o conteúdo da matéria por conta do semestre EARTE mais curto!))

4. **EXERCÍCIOS RESOLVIDOS**
    - Sincronização - Semáforos - Exercícios Resolvidos: Slides com vários exercícios resolvidos

5. **EXERCÍCIOS (valendo turings!!)**
    - **Responda o formulário fornecido juntamente com este roteiro**

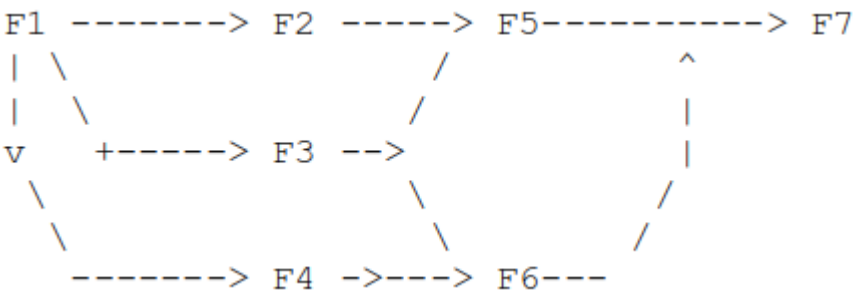
==========

Lista de Exercícios de Consolidação

O objetivo da lista é ajudar no estudo individual dos alunos. Soluções de questões específicas poderão ser discutidas em sala de aula, conforme interesse dos alunos.

==========

1. Considere o seguinte grafo de precedência:



que será executado por três processos, conforme código abaixo (primitivas “cobegin” e “coend” têm comportamento similar a “parbegin/parend”):

```
cobegin
P1: begin F1; F3; end;
P2: begin F2; F5; F7; end;
P3: begin F4; F6; end;
coend
```

Adicione e inicialize semáforos para este programa, e as respectivas chamadas às suas operações (down/up ou P/V), de modo que a precedência definida acima seja alcançada.

2. Considere o problema dos leitores e escritores, onde existem diversos processos que eventualmente fazem acessos de leitura a uma base de dados e diversos processos que eventualmente fazem acessos de escrita à mesma base. Vários acessos de leitura podem ocorrer simultaneamente, mas um acesso de escrita não pode ocorrer simultaneamente com acessos de nenhum tipo. Considere o código a seguir para os processos de leitura e de escrita. Suponha que todos os semáforos são iniciados com valor 1, e que a variável rc é inicializada em 0:

```
leitor:
  ...
1 while(1) {
2   P(R);
3   P(M);
4   rc++;
5   If (rc==1) P(W);
6   V(M);
7   V(R);
8   LÊ;
9   P(M);
10  rc--;
11  if (rc==0) V(W);
12  V(M);
13  //... consome
```

```
escritor:
  ...
1 while(1) {
2   //...produz
3   P(R);
4   P(W);
5   ESCREVE;
6   V(W);
7   V(R);
8 }
```

(a) Essa solução pode levar a *starvation* de escritores? (pode acontecer de um escritor nunca conseguir o acesso à base devido à seguida chegada de novos leitores?) Explique sua resposta, usando os números das linhas de código para se referir aos passos do programa.

(b) Explique o papel do semáforo M. Dê um exemplo de problema que poderia ocorrer caso as operações sobre ele fossem retiradas.

3. Um *semáforo múltiplo* estende semáforos de maneira a permitir que as primitivas *up* e *down* operem em vários semáforos simultaneamente. Isto é útil para requisitar e liberar vários recursos por meio de uma operação atômica. Como você implementaria tais primitivas usando os semáforos estudados no curso? (descreva em pseudo-código)

   - Observe que você, como implementador da primitiva, pode acessar o valor do contador associado ao semáforo diretamente. Ou seja, você tem disponível na hora de implementar down(S1, S2, ..., Sn) ou up(S1, S2, ..., Sn) uma rotina "valor_do_contador(Rx)", que devolve o valor do contador associado a um semáforo Rx.

```
boolean down(S1, S2, ..., Sn)
      down(mutex)
        if (S1.valor > 0) && (S2.valor > 0)...
      down(S1); down(S2); ...
            up(mutex)
            return true
       else
            up(mutex)
            return false
```

4. Resolva usando semáforos: "Três fumantes se encontram em uma sala com um vendedor de suprimentos para fumantes. Para preparar e usar um cigarro, cada fumante precisa de três ingredientes: tabaco, papel e fósforo, coisas que o vendedor tem à vontade no estoque. Um fumante tem o seu próprio tabaco, o segundo tem seu próprio papel, e o outro tem seu próprio fósforo. A ação se inicia quando o vendedor coloca à venda dois ingredientes na mesa, de forma a permitir que um dos fumantes execute esta prática dita como não muito saudável. Quando o tal fumante termina, ele acorda o vendedor, que escolhe então outros dois ingredientes (aleatoriamente) e coloca a venda, portanto desbloqueando outro fumante." *O Ministério da Saúde adverte: Fumar faz mal à Saúde!*

   Dica para a solução mais simples: O vendedor sabe que cada cliente tem consigo um dos ingredientes. Por exemplo, se ele coloca à venda tabaco e papel, ele sabe que tem um fumante que já tem o fósforo, e que este cliente está esperando justamente que o vendedor coloque tabaco&papel à venda.