

MAC0438 – Programação concorrente – 1s2014

EP3

Data de entrega: 12/06/2014

Prof. Daniel Macêdo Batista

1 Problema

Neste EP você deverá resolver o problema do *jantar dos selvagens*.

Uma tribo de $N > 1$ selvagens, cada um com peso $p_i (i \in 1, \dots, N)$, janta comendo a partir de um pote que tem capacidade para $C > 1$ porções de comida. Quando um selvagem quer comer ele se serve pegando 1 porção do pote, a não ser que o pote esteja vazio. Se o pote estiver vazio, o selvagem acorda 1 de $M > 1$ cozinheiros e espera o cozinheiro encher o pote com C porções de novo. Considere que o pote começa vazio e 1 cozinheiro aleatório enche o pote. O comportamento dos selvagens pode portanto ser resumido a:

```
while True:
    sirva-se do pote
    coma
```

E o comportamento do cozinheiro pode ser resumido a:

```
while True:
    durma
    encha o pote com C porções
```

Simule o jantar dos selvagens utilizando monitor. Além de seguir a descrição acima, a sua implementação deve garantir a justiça no acesso que os selvagens terão ao pote em duas situações:

- Situação uniforme: ao observar o jantar por um longo intervalo de tempo, as quantidades de porções comidas por cada selvagem devem estar bem próximas umas das outras.
- Situação com peso: ao observar o jantar por um longo intervalo de tempo, as quantidades de porções comidas por cada selvagem devem ser proporcionais aos pesos deles. Selvagens mais gordos devem comer mais do que selvagens mais magros.

Sua solução também deve garantir a justiça no acesso que os cozinheiros terão ao pote. Um mesmo cozinheiro não pode ser sempre o responsável por encher o pote. Observando o jantar por um longo intervalo de tempo, as quantidades de vezes que cada cozinheiro acordou devem estar bem próximas umas das outras.

Obs.: Sua solução não deve inserir restrições a mais no problema.

2 Requisitos

2.1 Monitor

Neste EP você deverá manipular monitores utilizando **apenas** as funções vistas em sala de aula com exatamente os mesmos nomes abaixo:

```
empty(cv), wait(cv), wait(cv,rank), signal(cv), signal_all(cv), minrank(cv)
```

Note que seu EP não precisa utilizar todas as funções (obviamente as funções `signal` e `wait` são obrigatórias). EPs que não implementem pelo menos as funções `signal` e `wait` receberão nota ZERO. Você deve indicar no LEIAME do seu EP onde estão as implementações de todas as funções que você implementou para manipular monitores.

2.2 Linguagem

Seu programa pode ser escrito em qualquer linguagem de programação desde que o compilador seja gratuito e tenha versão disponível para GNU/Linux, pois o EP será corrigido neste SO.

Algumas linguagens já possuem algumas estruturas voltadas para a implementação de programas com monitores. Leia a página da wikipedia a respeito disso:

```
https://en.wikipedia.org/wiki/Monitor\_\(synchronization\)#History
```

Caso você utilize alguma linguagem que já venha com as funções exigidas na subseção anterior implementadas, você pode utilizá-las, mas note que os nomes exigidos para as funções **devem** ser mantidos. Por exemplo, se em uma dada linguagem a função para acordar um processo da fila de uma variável de condição é `SignalZ`, você precisa criar um *alias*, uma outra função, etc... chamada `signal` que por sua vez chamará a `SignalZ`.

2.3 Entrada e Saída

O seu EP deverá receber na linha de comando os seguintes parâmetros nesta ordem:

```
arquivo  R  U|P
```

arquivo é o caminho para um arquivo texto no seguinte formato:

```
N
p1 p2 p3 p4 p5 p6 ... pN
C
M
```

onde N , p_i , C e M são as variáveis apresentadas no início deste documento. Elas possuem valores inteiros maiores que 1.

R é a quantidade de repetições que o EP deve executar. 1 repetição equivale a 1 pote finalizado pelos selvagens. Ou seja, se $R=1$, o EP vai começar com um cozinheiro aleatório enchendo o pote pela primeira vez e com os selvagens comendo este pote até o fim. Depois que o pote esvazia, o programa termina. Se R fosse 2, o selvagem que percebesse o pote vazio acordaria um cozinheiro e o programa terminaria depois que esse pote fosse esvaziado de novo.

Quando a opção U for passada na linha de comando, o EP deve executar a situação uniforme, e quando a opção for P ele deve executar a situação com peso, ambas explicadas no início deste documento.

O seu EP deverá imprimir na saída padrão as seguintes informações toda vez que um pote for esvaziado:

- Identificador do cozinheiro que encheu este pote;
- Identificador do selvagem que notou o pote vazio;
- Quantidade de vezes, atualizada incluindo as informações até este pote, que cada selvagem comeu;
- Quantidade de vezes, atualizada incluindo as informações até este pote, que cada cozinheiro encheu o pote;

3 Entregas

Você deverá submeter no paca um arquivo .tar.gz contendo os seguintes itens:

- fonte(s), Makefile (ou similar), arquivo LEIAME e um relatório em .pdf.

O desempacotamento do arquivo .tar.gz deve produzir um diretório contendo os itens. O nome do diretório deve ser ep3-membros_da_equipe. Por exemplo: ep3-joao-maria.

O relatório, que neste EP valerá 3,0, deve relatar o funcionamento do seu programa em quatro casos:

1. Entrada simples (poucos selvagens e poucos cozinheiros) na situação U
2. Entrada simples (poucos selvagens e poucos cozinheiros) na situação P
3. Entrada complexa (muitos selvagens e muitos cozinheiros) na situação U
4. Entrada complexa (muitos selvagens e muitos cozinheiros) na situação P

Note que dois arquivos devem ser usados. Um para os casos 1 e 2 e outro para os casos 3 e 4. O conteúdo do arquivo que você utilizou deve ser apresentado no relatório. Os valores dos parâmetros ficam ao seu critério. Note que o R precisa ser alto o suficiente para que os padrões esperados para a saída do programa apareçam.

Em cada caso você deve apresentar 2 gráficos e escrever um texto explicando esses gráficos, comentando se os resultados foram esperados e, se não foram os esperados, porquê.

Para o primeiro gráfico: um gráfico em linha em que o eixo x desse gráfico será o tempo lógico da simulação medida em quantidade de potes. O eixo y do gráfico vai ser a média do número de vezes que cada cozinheiro foi acordado. É de se esperar que o gráfico comece com um valor pequeno ($1/M$) e que esse valor vá aumentando com o passar do tempo.

Para o segundo gráfico: um gráfico em barras em que o eixo x desse gráfico deve ser o identificador de cada selvagem e o eixo y desse gráfico deve ser a quantidade de vezes que ele comeu, ou seja, a informação impressa pelo programa na última repetição. É de se esperar que no caso U as barras tenham os mesmos tamanhos e que no caso P as barras tenham tamanhos proporcionais aos pesos de cada selvagem.

O EP pode ser feito individualmente ou em dupla.