

**Enunciado do Trabalho para o Grau A – 2019/1****Simulador de Redes de Petri**

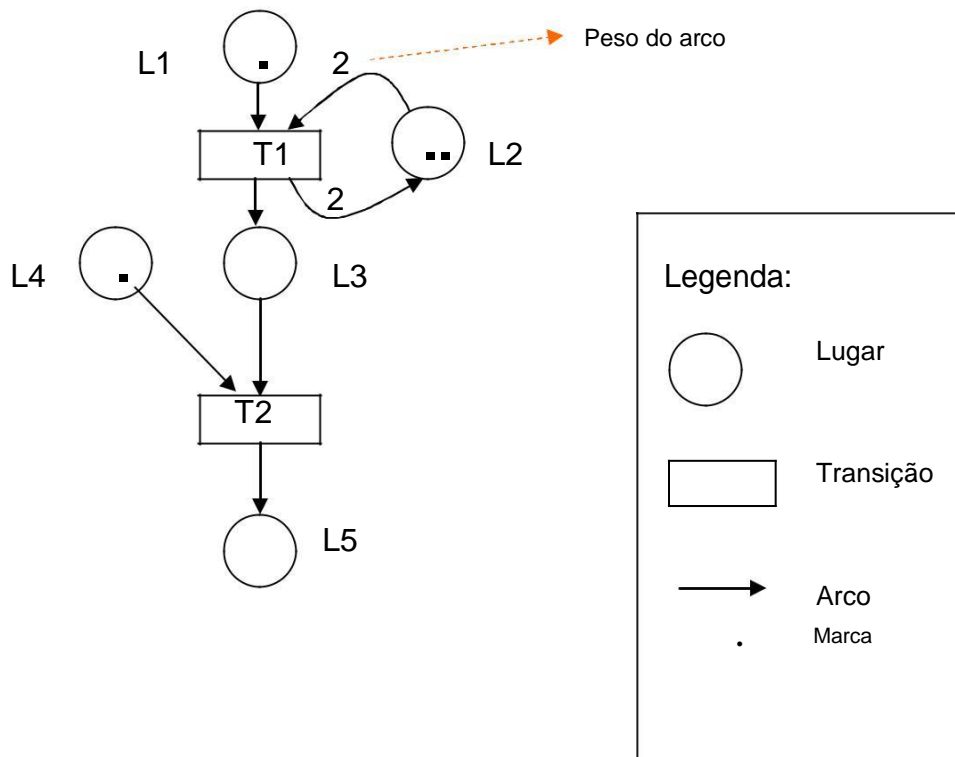
O programa deverá permitir a montagem e execução de uma rede de Petri do tipo lugar-transição temporizadas (Timed *Place-Transitions* Nets).

A rede é montada de maneira interativa com o usuário (ou a partir da leitura de um arquivo que contém a descrição da rede), permitindo que o usuário defina a quantidade de lugares e o número de transições que a rede conterá, bem como a marcação inicial (quantidade de marcas em cada lugar da rede) e o peso dos arcos que unem lugares e transições.

Quando da execução da rede, esta execução deverá ser passo-a-passo, e a cada ciclo de execução deverá ser mostrado quais as transições habilitadas para o próximo ciclo e o número de marcas em cada um dos lugares da rede.

O usuário deverá poder salvar a rede em um arquivo bem como realizar a leitura da rede a partir de um arquivo. Não é necessário o uso de interface gráfica; a apresentação dos resultados pode ser feita toda via console, na forma de tabelas.

Exemplo de Rede de Petri:



Obs.: na rede acima, no arco que vai de L2 para T1, o valor (peso) 2 indica que é necessário consumir 2 marcas de L2 para disparar T1 (desde que seja consumida também uma marca de L1); este valor é denominado de **Peso do Arco**.

## Comentários

1. a saída da execução do programa é textual; pode-se apresentar a informação através de tabelas. Exemplo:

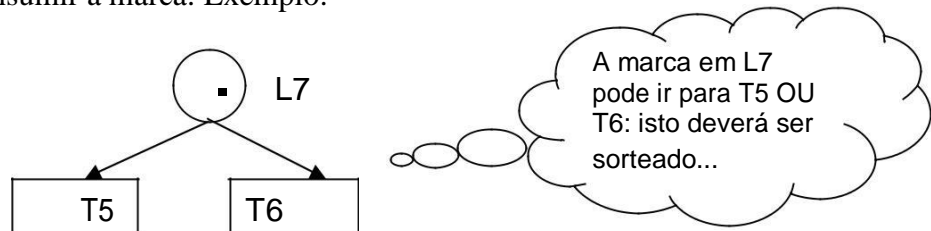
|          |   |   |   |   |   |
|----------|---|---|---|---|---|
| Lugar    | 1 | 2 | 3 | 4 | 5 |
| Marcação | 1 | 2 | 0 | 1 | 0 |

|              |   |   |   |
|--------------|---|---|---|
| Transição    | 1 | 2 | 3 |
| Habilitada ? | S | N | N |

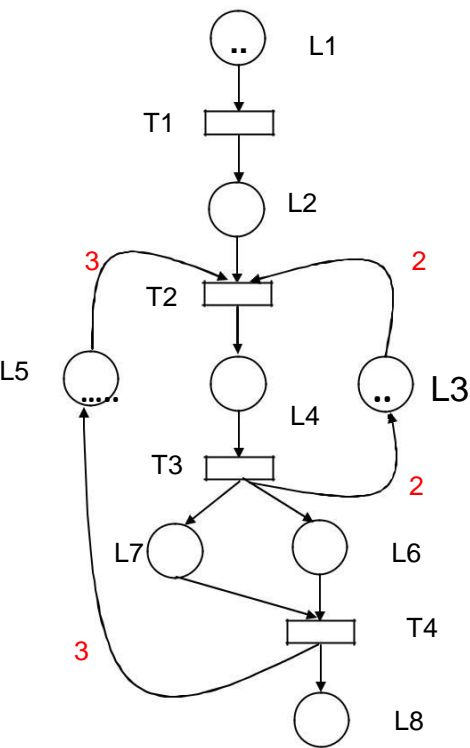
2. Critério para definir o disparo de transições: deve haver marcas **suficientes** em TODOS lugares de entrada de uma transição para esta estar habilitada. Quando a transição for disparada, as marcas são consumidas dos lugares de entrada e são enviadas marcas para os lugares de saída. A quantidade de marcas consumidas e enviadas depende do peso dos arcos que conectam lugares e transições. Um arco sem valor associado tem peso=1.
3. Cada ciclo de execução corresponderá a execução do disparo de todas transições habilitadas; o número do ciclo deverá ser mostrado na tela; para passar de um ciclo a outro (isto é, avançar a simulação) o usuário deverá apertar a tecla ENTER.
4. No início da execução do programa o usuário poderá fornecer os dados necessários para a montagem da rede de forma interativa (ou através da leitura de um arquivo que contenha esta descrição). Exemplo:

Quantos lugares: **3**  
 Quantas transições: **2**  
 Quais são os lugares de entrada de T1? **1, 3**  
 Quais são os lugares de entrada de T2? **2, 3**  
 Quantas marcas em L1 ? **10**  
 Quantas marcas em L2 ? **4**  
 Quantas marcas em L3 ? **0**  
 Qual o peso do arco de L1 para T1 ? **1**  
 Qual o peso do arco de L3 para T1 ? **2**  
 :

5. No caso de conflito/disputa por marca, deverá ser feito um sorteio para decidir qual transição irá consumir a marca. Exemplo:



Exemplo de rede e execução:



| Quantidade de marcas em cada lugar | Transição habilitada ? |
|------------------------------------|------------------------|
|------------------------------------|------------------------|

| Núm. do ciclo | L1 | L2 | L3 | L4 | L5 | L6 | L7 | L8 | T1 | T2 | T3 | T4 |
|---------------|----|----|----|----|----|----|----|----|----|----|----|----|
| 0 (inicial)   | 2  | -  | 2  | -  | 5  | -  | -  | -  | S  | N  | N  | N  |
| 1             | 0  | 2  | 2  | -  | 5  | -  | -  | -  | N  | S  | N  | N  |
| 2             | -  | 1  | -  | 1  | 2  | -  | -  | -  | N  | N  | S  | N  |
| 3             | -  | 1  | 2  | -  | 2  | 1  | 1  | -  | N  | N  | N  | S  |
| 4             | -  | 1  | 2  | -  | 5  | -  | -  | 1  | N  | S  | N  | N  |
| 5             | -  | -  | -  | 1  | 2  | -  | -  | 1  | N  | N  | S  | N  |
| 6             | -  | -  | 2  | -  | 2  | 1  | 1  | 1  | N  | N  | N  | S  |
| 7             | -  | -  | 2  | -  | 5  | -  | -  | 2  | N  | N  | N  | N  |



A implementação de aspectos de temporização é opcional para este trabalho.

Quanto a **temporização**: cada lugar pode ter um **tempo** associado. A unidade de tempo é fictícia (**Z**). A rede possui um “relógio global” que é inicializado com **relógio = 0<sub>Z</sub>** no início da simulação.

Exemplo: se  $Z(L_3) = 7_Z$  se uma marca chegar no lugar  $L_3$  com o relógio  $= 35_Z$  isso faz com que o tempo global salte para  $42_Z$  (pois  $35_Z + 7_Z = 42_Z$ ). A marca em  $L_3$  só fica efetivamente disponível para consumo de outra transição após esta passagem de tempo.

No final de **cada ciclo de simulação** o simulador deve apresentar o valor do relógio global.