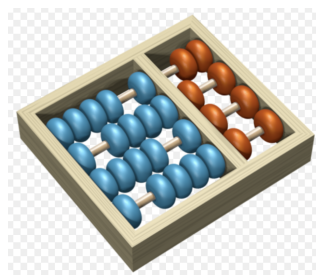


UNICAMP - UNIVERSIDADE ESTADUAL DE CAMPINAS

MARÇO 2021

MC504 - 1s2021

PROJETO 2 - O PROBLEMA DOS FUMANTES
RESOLVIDO COM SEMÁFOROS



Aluno: **Felipe Bueno MORET, RA: 155297**
Gabriel Kis Silvestre, RA: 155452
Professora: **Islene Calciolari Garcia**

Conteúdo

I	O Problema dos Fumantes	0
1	O que é o Problema dos Fumantes ?	0
II	Objetivo do Projeto	0
2	Usando semáforos para a solucionar o Problema dos Fumantes	0
3	Estratégia de solução	1
III	Resultados e Conclusão	2
4	Como é feita a representação do estado global ?	2
5	Como é feita a representação do ato de fazer o cigarro e fumar ?	3

Parte I

O Problema dos Fumantes

1 O que é o Problema dos Fumantes ?

O **Problema dos Fumantes**, proposto por **Suhas Patil** em 1971, é um problema de **concorrência** em Ciências de Computação. Inicialmente Suhas Patil pensou que o problema não poderia ser resolvido usando-se **semáforos**, porém o mesmo havia considerado restrições artificiais ao problema.

O problema consiste em **quatro threads**: **um produtor** e **três fumantes**. Os fumantes possuem uma vida infinita, sempre esperando por ingredientes, para em seguida fazerem e fumarem o cigarro. Os ingredientes necessários para a produção do cigarro são: **tabaco, papel e fósforo**.

Assumimos que o **produtor** possui uma quantidade infinita dos três ingredientes, e cada fumante possui uma quantidade infinita de um dos ingredientes, isto é, um fumante possui **tabaco infinito**, outro fumante possui **papel infinito** e por fim um terceiro fumante possui **fósforos infinitos**.

O **produtor** repetidamente escolhe dois ingredientes aleatoriamente, e torna-os disponíveis para os fumantes. Dependendo de quais ingredientes foram escolhidos pelo produtor, o fumante com o ingrediente faltante deve pegar os 2 ingredientes agora disponíveis e fazer o cigarro, em seguida avisando o **produtor** da falta de ingredientes novamente.

Esse problema representa um cenário muito importante em **sistemas operacionais**: o **produtor** representa o sistema operacional (devendo alocar recursos) e os **fumantes** representam aplicações que precisam de recursos. O problema está em dar certeza que se existirem recursos disponíveis que possibilitam o procedimento de uma aplicação, então tal aplicação deve ser "acordada". Ao mesmo tempo, gostaríamos de evitar "acordar" aplicações que ainda não possuem recursos para prosseguir.

Parte II

Objetivo do Projeto

2 Usando semáforos para a solucionar o Problema dos Fumantes

Para o presente projeto da **disciplina MC504**, nosso objetivo é usar o conceito de **semáforo** para tentar resolver o problema dos fumantes. Optaremos aqui pela versão do problema onde o código dos **produtores** não pode ser alterado - o que normalmente faria sentido em uma aplicação real, visto que não podemos pressupor uma alteração no código de um **SO** e também não podemos pressupor que uma *thread* tenha conhecimento sobre a execução de uma outra.

3 Estratégia de solução

Para resolver efetivamente o problema, faremos uso de semáforos, sendo eles:

1. Um semáforo para um **administrador de produção**, *production_Administrator*;
2. Um semáforo para controlar a **produção de tabaco**, *production_Tobacco*;
3. Um semáforo para controlar a **produção de papel**, *production_Paper*;
4. Um semáforo para controlar a **produção de fósforo**, *production_Match*.

Além disso, também estão presentes outros 3 semáforos auxiliares, chamados de **transportadores**, que avisam aos fumantes se eles podem começar a fazer um cigarro ou não. Os transportadores são representados pelos semáforos a seguir:

1. *transporter_Tobacco*;
2. *transporter_Paper*;
3. *transporter_Match*.

O comportamento desejado do produtor, isto é, um produtor que produz **dois ingredientes aleatórios a cada produção** é obtido através do uso de três threads que executam funções do tipo:

```
wait(production_Administrator());  
production_X.signal();  
production_Y.signal();
```

onde *X* e *Y* formam todas as combinações 2 a 2 (não ordenadas) dos ingredientes. Sendo assim, temos 3 threads para representar o produtor, representadas pelas funções **producing_X_Y()**.

Os fumantes são representados por threads que executam funções do tipo **smoking_with_X()**, para representar o ato de fumar tendo inicialmente o ingrediente $\mathbf{X} \in \{Tobacco, Match, Paper\}$.

Os transportadores decidem o valor dos semáforos **transporter_X**, através de threads que executam funções do tipo **transporting_X()** agindo da seguinte forma:

1. **transporting_X()** verifica se existe disponibilidade de algum outro ingrediente **Y** ou **Z**, e no caso de haver, ele informa o respectivo fumante capaz de produzir o cigarro indiretamente através de uma operação de **post** no semáforo do transportador de tal fumante;
2. No caso de não haver disponibilidade dos ingredientes **Y** ou **Z**, **transporting_X()** aumenta a disponibilidade do ingrediente **X** no sistema.

Parte III

Resultados e Conclusão

4 Como é feita a representação do estado global ?

O estado global é representado através de uma matriz que mostra a **disponibilidade de recursos** para cada participante: **agente (produtor) e fumantes com o ingrediente X**. Mais especificamente, a linha respectiva do agente indica os dois ingredientes **X** e **Y** que o produtor está produzindo em determinada iteração. Abaixo ilustramos tal matriz durante execução do programa:

```

      | M | T | P | S |
-----|---|---|---|---|
Agent | 1 | 1 | 0 | 0 |
Smoker Match | 1 | 0 | 0 | 0 |
Smoker Tobacco | 0 | 1 | 0 | 0 |
Smoker Paper | 0 | 0 | 1 | 0 |

----PRODUCTION OF TOBACCO AND MATCH IS GOING ON----

      /-| |
      /__| | TOBACCO & MATCH |
      | | |
      "-0-----0-0`         `0'0
```

Figura 4.1: Programa exibindo matriz de estado global do sistema: Na imagem, o produtor (agente) produz tabaco e fósforo, tendo a primeira linha da matriz o valor 1 nas coslunas de tais ingredientes.

5 Como é feita a representação do ato de fazer o cigarro e fumar ?

Assim que um fumante obtém os ingredientes para a produção do cigarro, ele fica **con-**
tente e é representado como na figura abaixo:

	M	T	P	S
Agent	0	0	0	0
Smoker Match	1	1	1	0
Smoker Tobacco	0	1	0	0
Smoker Paper	0	0	1	0

----SMOKER WITH MATCH IS MAKING THE CIGARETTE----

\o/
|
|
/ \
Match
■

Figura 5.2: Fumante inicialmente com fósforo contente por obter os ingredientes faltantes.

Assim que o cigarro fica pronto, finalmente o fumante fuma, sendo representado da seguinte maneira:

	M	T	P	S
Agent	0	0	0	0
Smoker Match	1	0	0	0
Smoker Tobacco	0	1	0	0
Smoker Paper	0	0	1	1

----SMOKER WITH PAPER IS SMOKING !!! :) ----

o~~
/|\
|
/ \
Paper
■

Figura 5.3: Fumante inicialmente com papel fumando.