



UNIVERSIDADE FEDERAL DE SANTA CATARINA CAMPUS TRINDADE
INE-DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
INE5410 - PROGRAMAÇÃO CONCORRENTE

Alunos: Caio Ferreira Cardoso, Gabriel Reimann Cervi, Lucas Brand Samuel
Martins,

Trabalho 2
Florianópolis 2023

Sumário

1 - Introdução

2 - Iniciando

3 - Implementação

4 - Sincronização

5 - Saída

6 - Erros

7 - Conclusão

Introdução

O código é um programa que lida com a fila de espera e funcionamento de uma atração, a Ixfera™, nela devem ser criadas threads que simulam as pessoas que estão indo presenciar a atração, além de threads que representam cada sessão da atração. O programa foi desenvolvido em Python usando os módulos de concorrência da linguagem.

Iniciando

Inicia-se o código recebendo 6 argumentos: o número de pessoas (`n_pessoas`), número de vagas dentro da Ixfera™ (`n_vagas`), a quantidade de unidades de tempo que as pessoas permanecem na atração (`permanencia`), o tempo máximo na unidade de tempo definida entre a chegada de duas pessoas na fila (`max_intervalo`), a semente usada para inicializar o gerador de número aleatórios (`semente`) e a quantidade de milissegundos que corresponde a uma unidade de tempo da simulação (`unid_tempo`).

Todos os valores devem ser maiores que zero, exceto a semente que é maior ou igual a zero.

Implementação

O projeto tem alguns tipos de threads, primeiro as threads que representam os clientes, as threads que representam as sessões da experiência (que também tem uma thread que a representa), uma thread responsável pela criação dinâmica das threads dos clientes e uma thread da Ixfera que controla a atração em geral.

As threads de sessões lidam obviamente com cada sessão da experiência, elas não têm um comportamento muito complexo, apenas “prendem” o cliente dentro da sessão até que o tempo definido seja alcançado. Além disso, registram seus tempos e somam eles ao final de suas execuções no contador geral de tempo. Ao final de sua execução avisa quando for pausada.

As threads cliente lidam com cada cliente, em sua execução o cliente é adicionado a uma sessão, espera ser liberado dela e avisa essa informação. Além disso, ele recorda quanto tempo ficou dentro da sessão, e adiciona isso ao contador do seu tipo de sessão (o tempo médio da sessão é dado numa média aritmética no final do código).

A thread da Ixfera, como já falado, controla a atração em geral, isto é, ela que cria as sessões, controla a entrada das pessoas e o término das experiências.

Existe também uma thread responsável pela criação das threads clientes, que cria novos clientes com o tempo de espera definido pelo usuário e insere eles na fila.

Sincronização

A sincronização do código ocorre através do uso de alguns mutexes e semáforos globais, bem como pertencentes às próprias pessoas. No ambiente global esses recursos de sincronização controlam o acesso à fila de pessoas, pessoas dentro da seção e os relógios que serão usados para calcular a taxa de ocupação. Enquanto isso, dentro de cada pessoa há semáforos que controlam o seus relógios pessoais baseados em quando elas entram e saem da atração.

Saída

A saída ocorre conforme o descrito no enunciado, com as seguintes possibilidades:

- “[Ixfera] Simulacao iniciada.”
- “[Ixfera] Iniciando a experiência A/B/C.”
- “[Ixfera] Simulacao finalizada.”
- “[Pessoa X / A/B/C] Aguardando na fila.”
- “[Pessoa X / A/B/C] Entrou na Ixfera (quantidade = Y).”
- “[Pessoa X / A/B/C] Saiu da Ixfera (quantidade = Y)”.

O primeiro grupo destas descreve as mensagens que lidam com o funcionamento do Ixfera em si e de suas sessões, indicando o início do programa, quando uma nova experiência é começada e o fim da experiência.

Já o segundo grupo trata da thread de cada pessoa, indicando quando chega na fila, quando entra na Ixfera e quando sai desta.

A ordem específica varia com cada semente utilizada para os números aleatórios e com a unidade de tempo que o usuário define.

Erros

O único erro identificado no código foi a necessidade de adicionar um tempo mínimo de espera no código da sessão, pois na ausência dele ocorre um erro, onde em alguns casos o cliente primeiro sai da experiência antes de entrar nela.

Conclusão

O grupo teve uma grande melhora em relação ao desempenho do primeiro projeto, conseguindo implementar a proposta por completo, além de desenvolver um entendimento muito melhor do funcionamento das ferramentas de concorrência.