

Relatório do Trabalho Final

Problemas de concorrência na 4ª Grande Guerra Ninja

Kleber Rodrigues da Costa Júnior

20/0053680

21 de janeiro de 2023

Universidade de Brasília – Instituto de Ciências Exatas

Departamento de Ciência da Computação – CIC0202 – Programação Concorrente

2022.2 – Turma A – Professor Eduardo Adilio Pelinson Alchieri

Prédio CIC/EST – Campus Universitário Darcy Ribeiro

Asa Norte 70919-970 Brasília, DF

kleberrjr7@gmail.com

Resumo: Insira aqui o resumo deste relatório.

Palavras-chave: Programação concorrente, condições de corrida, variáveis condicionais, locks, semáforos, ninjas, Naruto, algoritmo.

1. Introdução e explicação do problema

O problema proposto é ambientado no universo do anime Naruto.

A 4ª Grande Guerra Ninja chegou e Naruto, juntamente com seus aliados, precisa combater o grande vilão da história. Embora Naruto seja muito poderoso, alguns de seus amigos precisam de ajuda para conseguirem se tornar fortes o suficiente durante a batalha e conseguirem derrotar o inimigo.



Figura 1. Naruto envolto de poder.



Figura 2. O vilão mascarado.

Graças às habilidades de Naruto, ele consegue transferir seu poder para qualquer aliado que precise. Sabendo disso, seu amigo Rock Lee constantemente pede uma porção de poder à Naruto. Por estar ocupado com a batalha, Naruto não pode pessoalmente ajudar Lee, mas pode utilizar seu *jutsu clone das sombras* para fazer clones de si mesmo e os enviarem ao auxílio de seu amigo.



Figura 3. Jutsu clones das sombras em ação.



Figura 4. Rock Lee.

Toda vez que um clone da parte de seu poder a Lee, ele perde força vital e precisa recuperá-la com o auxílio de uma ninja médica. Entretanto, só há uma ninja médica no campo de batalha e ela precisa dar atenção especial aos demais ninjas feridos, só podendo revitalizar os clones quando nenhum ninja normal estiver precisando de ajuda. Seu nome é Sakura.



Figura 5. Clone dando poder ao Rock Lee.



Figura 6. Sakura curando Naruto.

Sakura utiliza sua reserva de *chakra* (poder dos ninjas) para curar feridos e revitalizar clones, porém só consegue atender 1 pessoa de cada vez. Além disso, toda vez que seu *chakra* se esgota ou fica muito baixo, ela deve parar com os atendimentos e reunir mais *chakra*.

Como organizar a Aliança Ninja para que Lee consiga lutar com o auxílio dos clones e nenhum ninja fique ferido por muito tempo?

2. Solução e implementação

Antes de nos aprofundarmos na solução, precisamos criar as threads que vão representar a abstração dos personagens. Como temos um número de clones e ninjas normais “indefinido” (o número está definido nas constantes `NUM_CLONES` e `NUM_REG_NINJAS`, respectivamente), criaremos um vetor de threads para cada grupo. Já nossa ninja médica Sakura e o nosso amigo Rock Lee só precisam de uma thread para representa-los.

-- POSSÍVEL FOTO --

Deste ponto em diante, considere todos os mecanismos utilizados (locks, variáveis de condição e semáforos) devidamente declarados e inicializados, pois não iremos mostrar esses trechos a fim de economizar páginas.

Abordando a primeira parte do problema: precisamos garantir que os ninjas normais e os clones possam ser curados pela mesma ninja médica sem problemas de concorrência. O recurso compartilhado em questão é o *chakra* de Sakura, que é drenado para curar os feridos e revitalizar os clones, nesse caso a zona de cura é uma região crítica e deve ser protegida com um lock para garantirmos a exclusão mútua.

-- POSSÍVEL FOTO --

Sakura só pode fazer 2 coisas: reunir/recarregar *chakra* e curar os feridos, logo precisamos estabelecer as condições que levarão ela a executar uma tarefa ou outra. Para curar um ninja Sakura gasta 25% de seu *chakra* e para revitalizar um clone ela gasta 15% de seu *chakra*. Como ninjas são prioridade, enquanto Sakura tiver um nível de *chakra* acima de 25%, ela continuará com seus cuidados médicos dormindo na condição e liberando o lock da zona de cura, caso contrário ela deverá parar os atendimentos para recarregar seu nível de *chakra* e após isso enviar um sinal para todos os ninjas e clones esperando seu retorno.

É importante que utilizemos um `pthread_cond_broadcast`, pois temos mais de uma thread esperando e todas elas precisam competir pelo recurso.

-- POSSÍVEL FOTO --

Os ninjas normais “utilizam” o chakra de Sakura para se curarem, e portanto devem acessar esse recurso dentro do mesmo lock utilizado anteriormente. Por, terem prioridade, sempre que conseguirem entrar na região crítica incrementaremos uma variável para indicar que um ninja ferido está precisando de atendimento. Uma vez dentro da zona de cura, o ninja só pode fazer 2 coisas: ser curado ou esperar. Ele só vai esperar caso Sakura não tenha chakra suficiente para lhe curar, nesse momento ele deve avisar (acordar) a ninja médica para que ela vá reunir mais chakra (`pthread_cond_signal`, pois só há 1 Sakura). Caso contrário, ele irá decrementar o recurso compartilhado retirando a quantidade de chakra que precisa para ser curado, além da variável indicadora de ninjas feridos, e retornará para o campo de batalha liberando o lock da zona de cura.

-- POSSÍVEL FOTO --

Assim como os ninjas e a Sakura, os clones precisam acessar a região crítica através do lock da zona de cura, uma vez dentro eles podem as mesmas 2 coisas mencionadas no parágrafo anterior para os ninjas: ser revitalizado e esperar. Entretanto, os clones não possuem prioridade no atendimento, portanto as condições de parada devem ser:

- a. Sakura está com o nível de chakra abaixo de 15% (necessário para revitalizar o clone) ou
- b. Há ninjas feridos esperando atendimento

Caso qualquer uma dessas condições seja verdadeira, o clone irá esperar em `pthread_cond_wait(&naruto_clone_cond, &healing_zone)`, liberando o lock da zona de cura e acordando a Sakura. Em determinado momento, é possível que o clone acorde Sakura no momento em que ela já está acordada, mas isso não causa nenhum tipo de deadlock devido à condição de parada de Sakura. Sakura só vai “dormir” quando seu nível de chakra for inferior à quantidade necessária para curar um ninja, logo se houver ninjas esperando por atendimento, então certamente Sakura estará juntando mais chakra, se houver ninjas ativos, então eventualmente um deles irá acordar Sakura. Os clones não causam deadlock.

Se nenhuma condição for satisfeita, o clone é revitalizado, ganhando 15 pontos de poder vital e liberando o lock da zona de cura.

-- POSSÍVEL FOTO --

Para doar uma porção de poder a Rock Lee, cada clone precisa de 40 pontos de poder vital, logo testamos em loop infinito quantos pontos o clone possui. Nesse momento utilizaremos um semáforo que serve como contador para as porções de poder dadas à Lee. Toda vez que um clone junta 40 pontos ou mais, ele incrementa as permissões do semáforo com um `sem_post(&power_portions)`.

-- POSSÍVEL FOTO --

A função de Rock Lee é simples: ele irá utilizar uma porção de poder sempre que estiver disponível, isto é, sempre que o semáforo tiver permissões. Estas permissões serão

incrementadas sempre que um clone doar uma porção de poder, como mencionado anteriormente. Sobre o semáforo, ele deve ser iniciado com 0 permissões, pois inicialmente Lee não tem a capacidade de lutar e deve ficar bloqueado.

-- POSSÍVEL FOTO --

3. Análise de resultados

Insira aqui a análise dos resultados.

4. Conclusão

Insira aqui a conclusão.

Referências Bibliográficas

Insira aqui as referências bibliográficas.