

Introdução

A programação concorrente é um paradigma de programação usado na construção de programas que fazem uso da execução simultânea de diversas tarefas que podem ser implementadas como programas separados ou como um único programa que dispara várias linhas de execução em paralelo. As linhas de execução também são conhecidas como threads.

A programação concorrente permite a execução em paralelo de vários programas no computador e está relacionada com a programação paralela.

As tarefas sendo podem ser executadas concorrentemente em um único processador, vários processadores em um equipamento ou vários processadores espalhados em uma rede de servidores.

Vantagens da Programação Concorrente

A principal vantagem do uso da programação concorrente é o aumento do desempenho dos programas, pois é possível aumentar a quantidade de tarefas executadas em um determinado período de tempo.

Por exemplo, em uma grande empresa, pode-se calcular o faturamento mensal da empresa executando o cálculo do faturamento de forma concorrente, processando várias filiais simultaneamente. Neste caso, a programação concorrente permite ganho de tempo pois processamos todas as filiais ao mesmo tempo, ao invés de processar o faturamento de uma filial de cada vez. Esta abordagem é muito utilizada profissionalmente.

Outra vantagem é que o projeto dos programas se torna mais simples com relação à concorrência entre os programas pelos recursos compartilhados.

Desafios da Programação Concorrente

O grande desafio da programação concorrente é o compartilhamento de recursos, a comunicação e a interação entre os programas que são executados concorrentemente. O objetivo deve ser o de administrar o acesso concorrente aos recursos computacionais, tais como por exemplo, o acesso aos discos ou a outros recursos compartilhados, como a impressora.

Comunicação e Interação entre programas concorrentes

Em um sistema computacional concorrente, os programas executados de forma concorrente podem ter a necessidade de se comunicarem para troca de informações. Neste contexto, é importante que o programador possa ter alguma informação sobre como é feita a comunicação entre os programas executados concorrentemente. Esta informação sobre a comunicação entre programas concorrentes é importante para que se evite que um programa interfira em outro programa.

Conflitos e Compartilhamento de Recursos

É preciso evitar conflitos entre os programas concorrentes. Os conflitos podem ocorrer porque os programas concorrentes fazem uso de recursos compartilhados e por isso a necessidade de se ter controle de concorrência.

Em alguns sistemas computacionais concorrentes a comunicação entre os processos concorrentes é 'escondida' do programador mas em outros sistemas a comunicação é explícita, podendo ser interpretada pelo programador. Existem dois tipos de comunicação explícita para a programação concorrente: a comunicação por memória compartilhada e a comunicação por troca de mensagens.

Comunicação por troca de mensagens

Este tipo de comunicação consiste na troca de mensagens entre processos sendo executados concorrentemente através da chamada de funções ou serviços. Caso os processos estejam sendo executados em máquinas diferentes é a única forma de comunicação possível. Baseia-se na colaboração entre os processos, onde para uma mensagem enviada por um processo sempre deve haver uma resposta pelo outro processo.

Comunicação por memória compartilhada

Este tipo de comunicação os processos comunicam-se através do acesso à áreas de memória que são compartilhadas entre os processos concorrente. Neste tipo de comunicação deve ser feito gerenciamento do uso da memória compartilhada a fim de que um processo não interfira na execução de outro processo sendo executado concorrentemente.

A comunicação por troca de mensagens é mais simples e mais eficiente do que a comunicação por memória compartilhada.

Suporte à programação concorrente

São poucas as linguagens com suporte nativo à programação concorrente. Atualmente apenas JAVA e C# possuem recursos que permitem a construção de programas concorrentes. Destas duas linguagens, JAVA se destaca por ter portabilidade a outros sistemas operacionais ao contrário de C# que é restrita aos sistemas Windows.

As linguagens C e C++ oferecem suporte à programação concorrente através de bibliotecas de funções para programação concorrente.

Linguagens específicas para Programação Concorrente

Existem ainda as linguagens que foram projetadas especificamente para programação concorrente. A linguagem Erlang é muito utilizada atualmente. A principal vantagem de Erlang é suporte a concorrência. Tem um pequeno, mas poderoso, conjunto de funções primitivas para criar processos e fazer com que eles se comuniquem. Processos são o principal meio de criar uma aplicação em Erlang.