**“Uma entrada de dados muito grande pode não ser muito eficaz utilizando um algoritmo e mais eficaz usando outro algoritmo”**

Medir a eficiência de um software em termos de tempo é muito complexo uma vez que temos muitas variáveis envolvidas.

Por exemplo:

* Hardware
* O algoritmo
* A linguagem
* Entre outros

É necessária uma medida que indique qual algoritmo é mais eficiente.

Com isso, medir o tempo pode não ser muito adequado

**IPC – Interprocess Communication**

É o fato de eu ter processos no sistema operacional que precisam disputar um determinado recurso compartilhado na CPU.

Condição de corrida é quando vários processos estão disputando um recurso.

**O problema da coerência:** quando duas ou mais tarefas acessam simultaneamente um recurso compartilhado, podem ocorrer problemas de consistência dos dados ou do estado do recurso acessado.

**Região crítica:** é a área de um código ou recurso compartilhado que depende expressivamente que o acesso a mesma seja realizado de maneira sequencial.

Exemplo: se uma variável x deve ser lida pelo processo A e o processo B precisa gravar na mesma variável x ao mesmo tempo, o processo A pode obter o valor antigo ou o novo valor de x.

Em casos como esses, uma seção crítica é importante. No caso acima, se A precisar ler o valor atualizado de x, executar o processo A e o processo B ao mesmo tempo pode não dar os resultados necessários.

Para evitar isso, a variável x é protegida por uma seção critica. Primeiro, B obtem o acesso a seção. Assim que B terminar de escrever o valor, A terá o acesso a seção critica e a variável x poderá ser lida.

Uma seção critica geralmente é utilizado quando um programa multi-thread deve atualizar várias variáveis relacionadas sem que um thread separado faça alterações conflitantes nesses dados.

**O que são threads?**

É uma forma de um processo dividir a si mesmo em duas ou mais tarefas, podendo executar elas concorrentemente.

Em um programa single thread, inicia a etapa 1 e continua sequencialmente (etapa 2, etapa 3 e assim por diante) até atingir a etapa final. Aplicações multi-thread permitem que você execute varias threads ao mesmo tempo, cada uma executando um passo por exemplo.