**Capítulo 1 - Computer Abstractions and Technology**

**1.1 Introduction:**

**PC (personal computer):** computador criado para uso individual; normalmente incorpora display gráfico, mouse e teclado.

**Slide 1**

**Sistemas Cyber-Physical:** consiste na combinação de um componente de software com partes mecânicas ou eletrônicas; o controle, o monitoramento, a transferência de dados e o intercâmbio de dados são geralmente executados via internet em tempo real.

**Componentes de um Computador:**

**Disco:** guarda informações que não estão em uso no momento; altíssima capacidade de armazenamento; não é volátil (os dado não se perdem); tempo de acesso bem maior que o das memórias.

↪ *disco rígido*: não pode ser removido do gabinete do computador.

↪ *disco flexível*: pode ser removido do gabinete e levado para qualquer lugar.

**Memória:** guarda informações que estão em uso no momento; acesso rápido e fácil; é volátil (pode ser facilmente perdida caso a máquina seja desligada).

↪ *memória cache:* grupo de chips de memória muito rápidos que aceleram o processamento; baixa capacidade.

↪ *memória RAM:* pequena e rápida para acesso imediato aos dados.

↪ *DRAM:* precisa que a informação seja atualizada o tempo todo para que que permaneça armazenada.

↪ *SRAM:* consegue manter os dados mesmo sem atualização contínua, os mesmos são perdidos somente após a interrupção da fonte de energia.

**Unidade de Controle - CPU:** responsável direto pela execução das instruções de um programa; realiza o processamento das informações e controla a sequência de dados, a memória, etc.

**Entrada/Saída:** teclado, mouse, monitor de vídeo, joystick, etc; uma característica comum é a baixa velocidade.

**Desempenho**

O **desempenho/performance** do hardware é responsável pela eficácia de todo o sistema.

Como o desempenho pode ser definido de várias maneiras, para diferentes tipos de aplicação, deverão ser empregadas métricas diferentes da performance.

**Tempo de Resposta/Execução:** o tempo decorrido entre o início e o final do seu programa; total de tempo necessário para um computador completar uma tarefa, incluindo acesso ao disco, acesso à memória, atividades de entrada/saída, etc...

**Throughput/Taxa de Execução:** a quantidade total de trabalho executado em um determinado intervalo de tempo; observado pelo desenvolvedor.

Diminuindo o tempo de execução, conseguimos melhorar a taxa de execução, por exemplo: usando uma versão mais rápida de um processador, precisamos de menos tempo para executar uma tarefa, o que aumenta o número de tarefas executadas por unidade de tempo.

Aumentando a taxa de execução, conseguimos diminuir o tempo de execução, por exemplo: aumentando o número de processadores em um multiprocessador faz com que mais tarefas possam ser executadas em paralelo, mesmo o tempo de execução individual de tarefas não sendo alterado, o tempo de espera em fila de agendamento diminui, diminuindo também o tempo de resposta.

**❊** **Calculando o desempenho da máquina X:**

**❊ Máquina X é n vezes mais rápida do que a máquina Y:**

O tempo de resposta é um número útil, porém muitas vezes não é bom para fins de comparação. O principal foco para medida de desempenho é a CPU Execution Time.

**CPU Execution Time**: consiste no tempo gasto para executar as instruções do programa; não conta o tempo de espera para escalonamento de E/S nem de execução de outros programas.

**O CPU Execution Time é medido por:**

Para melhorar o desempenho nós precisamos reduzir o número de ciclos de clock exigido pelo programa ou reduzir o período do clock/aumentar a frequência do clock.

Instruções podem usar um número diferente de ciclos para executar.

**CPI (cycles per instruction):** um número médio de ciclos de clock por instrução.

É importante ter em mente que alterar o tempo do ciclo, muitas vezes, requer alteração no número de ciclos necessários para várias instruções.

**❊** Para calcular a quantidade de ciclos de uma CPU:

**❊** Equação do Desempenho:

Diferentes tipos de instruções podem possuir CPI’s diferentes:

Seja CPI1 = ciclos por instrução para Classe 1 de instruções.

Seja C1 = número de instruções para Classe 1 de instruções.

**MIPS (Millions Instructions Per Second):**  algumas vezes usada como métrica de desempenho; máquinas rápidas possuem um valor de MIPS alto.

**3 problemas** causados por usar a métrica MIPS como medidora de desempenho:

↪ não leva em conta a capacidade de instruções; não é possível usar o MIPS dos computadores para comparar conjuntos de instruções diferentes.

↪ MIPS varia entre programas de um mesmo computador.

↪ MIPS pode variar inversamente com o desempenho; uma classificação elevada do MIPS nem sempre significa melhor desempenho.

**Tempo de execução do processador**: é o tempo gasto por ele na execução das instruções dedicadas a um programa e não inclui o tempo gasto com entrada/saída, nem o tempo gasto para a executar outros programas diferentes, no caso de sistemas que compartilham o tempo do processador.

**Benchmarks:** são programas desenvolvidos especialmente para trabalhar na medida da performance; os melhores programas para se escolher como benchmarks são programas reais.

**SPEC (System Performance Evaluation Corporation)**: corporação formada para estabelecer benchmarks e ferramentas para avaliar performance.

**Conclusão**: desempenho é específico de cada programa.

**Para causar melhorias no desempenho:**

↪ *aumento da frequência do clock* (sem aumentar CPI);

↪ *melhorias na organização do processador* que diminua a CPI;

↪ *melhorias no compilador* que reduza a CPI ou o número de instruções;

↪ *escolhas de algoritmo ou da linguagem* que afete o número de instruções.