


GBC036—Arq. e Org. de Computadores I



## Avaliando o Desempenho de UCPs

---

Universidade Federal de Uberlândia  
Faculdade de Computação  
Prof. Dr. rer. nat. Daniel D. Abdala

## Nesta Aula

---

- Definição de desempenho;
- Mensuração de desempenho;
- Desempenho da UCP e seus fatores;
- Desempenho de instruções;
- Considerações acerca do consumo de energia;
- Desempenho em sistemas multiprocessados;
- Benchmarking;

## A Importância do Desempenho

---

- Avaliar desempenho de computadores
  - Desafiador / Complexo;
  - Importante → a classe de problemas resolvíveis é limitada pelo desempenho;
  - Escolha do sistema computacional;
  - Estimar as necessidades computacionais para uma determinada aplicação;

## O que é Desempenho?

---

- O que significa dizer?  
“Computador A possui melhor desempenho que o Computador B”
- Uma analogia para definir o termo:

## Exemplo

Veículo	Vel. média	Alcance	Capacidade	Flexibilidade
Trem	100km/h	4.000 km	600	Baixa
Avião	800km/h	8.000 km	400	Baixa
Carro	90km/h	600 km	5	Alta
Ônibus	70km/h	700 km	42	Média

- Qual veículo apresenta melhor desempenho?
  - Depende de como definimos desempenho:
    - Velocidade
    - Alcance
    - Capacidade
    - flexibilidade

## Do ponto de vista computacional...

---

- Um programa em dois computadores distintos:
  - Computador que termina o programa antes possui maior desempenho
- Datacenter com diversos computadores rodando diversas tarefas:
  - O computador mais rápido é aquele que completa o maior número de tarefas por dia.

## Throughput & Tempo de Execução

- Largura de Banda
  - Ou Throughput;
  - Quantidade total de trabalho em um determinado tempo;
- Tempo de Resposta
  - Ou tempo de execução;
  - Tempo decorrido entre o início e o término de uma tarefa;

7

## Questões Chave:

- Substituir o processador em um sistema computacional por um processador mais rápido.
- Acrescentar processadores adicionais a um sistema que utiliza múltiplos processadores para tarefas separadas.

8

## Tempo de Resposta

- Desempenho é inversamente proporcional ao tempo de resposta
- Considerando dois computadores, dizemos que se

$$Desempenho_x = \frac{1}{Tempo\ de\ Execução_x}$$

$$Desempenho_x > Desempenho_y$$

$$\frac{1}{Tempo\ de\ Execução_x} > \frac{1}{Tempo\ de\ Execução_y}$$

9

## Desempenho Relativo

- Computador A executa um programa em 10s e computador B em 15s. Quanto A é mais rápido que B?

$$\frac{Desempenho_x}{Desempenho_y} = n$$

$$\frac{15}{10} = 1,5$$

10

## Outros pontos a serem considerados

- **Tempo de execução** → tempo decorrido entre o início de um processo e seu término.
- **Tempo de CPU** → tempo efetivamente transcorrido em que o processador trabalha no processo.
- Tempo de acesso a memória
- Tempo de entrada e saída

11

## Desempenho da UCP

- Considere um programa executando em um computador hipotético qualquer:

$$Tempo\ de\ Exec.\ da\ CPU = \#Ciclos\ de\ Clock \times Tempo\ do\ ciclo\ de\ clock$$

- Fica claro que duas formas de se melhorar o tempo de execução e consequentemente o desempenho seriam:
  - Diminuir o número de clocks necessários para executar o programa;
  - Diminuir a duração do ciclo de clock.

12

## Desempenho da Instrução

- A equação anterior não faz menção ao # de instruções que compõem o programa;
- Podemos definir o tempo de execução de um programa como:
  - # de instruções x tempo médio das instruções
- CPI → Ciclos de Clock por Instruções (Clock cycles Per Instruction)
- Consequentemente:  
 $CPI = \#instr. \times \# \text{ médio de ciclos de clock por instr.}$

13

## Equação Clássica de Desempenho da UCP

$$Tempo_{UCP} = \frac{\#instruções \times CPI}{Taxa \text{ de Clock}}$$

- Exercícios

14

## Desempenho em Sistemas Paralelos

- Intuição
  - 1  $\mu$ Proc | 1 thread → tempo de execução = x ns
  - 2  $\mu$ Proc | 1 thread → tempo de execução = x/2 ns
  - 4  $\mu$ Proc | 1 thread → tempo de execução = x/4 ns
- Certo?
  - Infelizmente ... NÃO!!!!

15

## Lei de Amdahl

- Frequentemente usado em computação paralela para prever o máximo **speedup** teórico usando múltiplos processadores;

$$s(n) = \frac{1}{(1 - B) + \frac{B}{n}}$$

- $n$  = número de threads
- $B$  = parcela do algoritmo puramente sequencial [0,1]

16

## Bibliografia

- Livro do Patterson, edição nova. Páginas 19 a 29

17

## Trabalho Extra

- Escreva um pequeno artigo no qual a lei de Amdahl é revisada e explicada. Aponte os problemas com ela e então correlacione as correções previstas pela lei de Gustafson;
- 2 pontos na média;
- Entrega no final do semestre;

18