

# Desempenho

- Os três processadores possuem o mesmo conjunto de instruções
- Qual tem o melhor desempenho?
- Se em cada um, cada programa demora 10 seg. para executar, qual a qtde de instruções?

Processor	Clock rate	CPI
P1	2 GHz	1.5
P2	1.5 GHz	1.0
P3	3 GHz	2.5

$$\text{CPU time} = \text{Instruction count} \times \text{CPI} \times \text{Clock cycle time}$$

$$\text{CPU time} = \frac{\text{Instruction count} \times \text{CPI}}{\text{Clock rate}}$$

# Desempenho

- Desempenho: considerando I qtde de instruções
  - P1: 2ghz  $\rightarrow 5 \times 10^{-9}$  s  $\rightarrow$  CPU Time =  $I * 1.5 * 5 \times 10^{-9}$  s =  $I * 7,5 \times 10^{-9}$  S  $\rightarrow$   **$I * 7,5E-10$**
  - P2: 1.5 GHz  $\rightarrow 2 \times 10^{-9}$  s  $\rightarrow$  CPU Time =  $I * 1.0 * 2 \times 10^{-9}$  s =  $I * 2 \times 10^{-9}$  S  $\rightarrow$   **$I * 2E-10$**
  - P3: 3 GHz  $\rightarrow 3.3 \times 10^{-9}$  s  $\rightarrow$  CPU Time =  $I * 2.5 * 3.3 \times 10^{-9}$  s =  $I * 8,25 \times 10^{-9}$  S  $\rightarrow$   **$I * 8,3E-10$**
  - ***P2 tem menor CPU Time, de  $I * 2E-10$ .***

# Desempenho

- Considerando cada programa demora 10 segundos
- P1:  $10 = I * 7,5E-9 \text{ S} \rightarrow I = 1E10$
- P2:  $10 = I * 2E-9 \text{ S} \rightarrow I = 10E10$
- P3:  $10 = I * 8,3E-9 \text{ s} \rightarrow I = 1E10$
- ***No mesmo intervalo de tempo, P2 executa uma qtd de instruções 10 vz maior que P1 e P3***

# Desempenho

- Considere dois processadores P1 e P2, que implementam o mesmo conjunto de instruções, divididos nas classes A,B,C,D.
- P1 e P2 tem os seguintes clocks e CPIs médio para cada classe:

	Clock rate	CPI Class A	CPI Class B	CPI Class C	CPI Class D
P1	1.5 GHz	1	2	3	4
P2	2 GHz	2	2	2	2

# Desempenho

- Considere um programa que tenha  $10E6$  instruções, sendo 10% classe A, 20% classe B, 50% classe C e 20% classe D. Qual processador será mais rápido?
- Qual clock cycle?

	Clock rate	CPI Class A	CPI Class B	CPI Class C	CPI Class D
P1	1.5 GHz	1	2	3	4
P2	2 GHz	2	2	2	2

# Desempenho

- Primeiro identificar o *clock cycle* geral

$$\text{CPU clock cycles} = \sum_{i=1}^n (\text{CPI}_i \times C_i)$$

- $\text{CPI}_i$  de cada classe,  $C_i$  qtde de instruções
- P1:  $1 \times (0,10 \times 10\text{E}6) + 2 \times (0,20 \times 10\text{E}6) + 3 \times (0,50 \times 10\text{E}6) + 4 \times (0,20 \times 10\text{E}6) = 28 \times 10^6$  **ciclos de clock**
- P2:  $2 \times (0,10 \times 10\text{E}6) + 2 \times (0,20 \times 10\text{E}6) + 2 \times (0,50 \times 10\text{E}6) + 2 \times (0,20 \times 10\text{E}6) = 20 \times 10^6$  **ciclos de clock**

# Desempenho

- P1: clock= 1.5GHz, rate=  $700 \times 10^{-12}$  s (pico segundos)
- P1:  $28 \times 10^6$  ciclos de clock \*  $700\text{E-}12 = 18\text{E-}3$  s
- P2: clock = 2GHz, rate=  $500 \times 10^{-12}$  s (pico segundos)
- P2:  $20 \times 10^6$  ciclos de clock \*  $500\text{E-}12 = 10\text{E-}3$  s
- ***P2 possui o melhor desempenho. P2 é 1,8 vezes mais rápido que P1 para o programa escolhido.***

# Desempenho

- Considere um programa que utilize as seguintes quantidades de instruções:

Arith	Store	Load	Branch	Total
500	50	100	50	700

- Assumindo que instruções do tipo Arith gastam 1 ciclo de clock, load e store gastam 5 ciclos e branch gastam 2 ciclos, qual o tempo de execução em um processador de 2GHz?
- Qual o CPI de todo programa?
- Se acontecer uma melhoria de 50% na velocidade das instruções load e store, qual o speed up e o CPI final?



# Desempenho

- Ciclos de clock =  $500 \times 1 + 50 \times 5 + 100 \times 5 + 50 \times 2 = 1350$ . Tempo total =  $1350 \times 500\text{E-}12 \text{ s} = 6,75\text{E-}9 \text{ s}$ . (6,75 nanoS)
- CPI Médio do programa =  $1350/700 = 1.9$
- Melhorando load e store em 50%, CPI = 2 ciclos de clock:
  - C.C. =  $500 \times 1 + 50 \times 2 + 100 \times 2 + 50 \times 2 = 900$ .  
Tmp =  $900 \times 500\text{E-}12 = 4,45 \text{ nanoS}$ .
  - CPI Médio do programa =  $900/700 = 1.3$
- ***Speedup =  $6,75/4,45 = 1.5$  vezes mais rápido***

# Desempenho

- Lei de Amdahl: medida do impacto de determinada melhoria no desempenho

$$\text{Execution time after improvement} = \frac{\text{Execution time affected by improvement}}{\text{Amount of improvement}} + \text{Execution time unaffected}$$

- Identifica o limite de determinada melhoria.

# Desempenho

- Outra métrica: MIPS
  - Usar somente o clock é uma métrica ruim de comparação
  - Outra métrica ruim é o MIPS: Milhoes de instruções por segundo.
  - Fácil de compreender: computador mais rápido possui MIPS maior....

# Desempenho

- Contudo:
  - MIPS não leva em conta a capacidade das instruções. Dois computadores com instruções diferentes não podem ser comparados pelo MIPS.
  - MIPS pode variar com programas diferentes dentro de um mesmo computador. Um programa pode usar instruções que tem CPI maior ou menor

	Clock rate	CPI Class A	CPI Class B	CPI Class C	CPI Class D
P1	1.5 GHz	1	2	3	4
P2	2 GHz	2	2	2	2

# Desempenho: resumo

- Componentes da medida de desempenho

Componentes	Unidade
Tempo de execução na CPU	Segundos para o programa
Contagem de instruções	Instruções executadas no programa
CPI	Número média de ciclos de clock por instrução
Ciclo do clock	Segundos para ciclo do clock

# Desempenho: resumo

- Algoritmo: A escolha do algoritmo afeta a contagem das instruções de máquina gerada e o CPI.
- Um algoritmo que usa ponto flutuante, irá gerar uma instrução cuja qtde de ciclo de clock seja maior do que uma operação de cálculos sobre inteiros

# Desempenho: resumo

Linguagem de Programação: O tipo da linguagem implica na geração de mais instruções para a CPU.

Quanto maior a abstração dos dados de uma linguagem, maior o CPI, devido as chamadas indiretas aos dados.

# Desempenho: resumo

- Compilador: a eficiência do compilador afeta na quantidade final de instruções geradas.
- Além disso, existem várias estratégias de otimização que o compilador pode usar para diminuir o CPI médio do programa

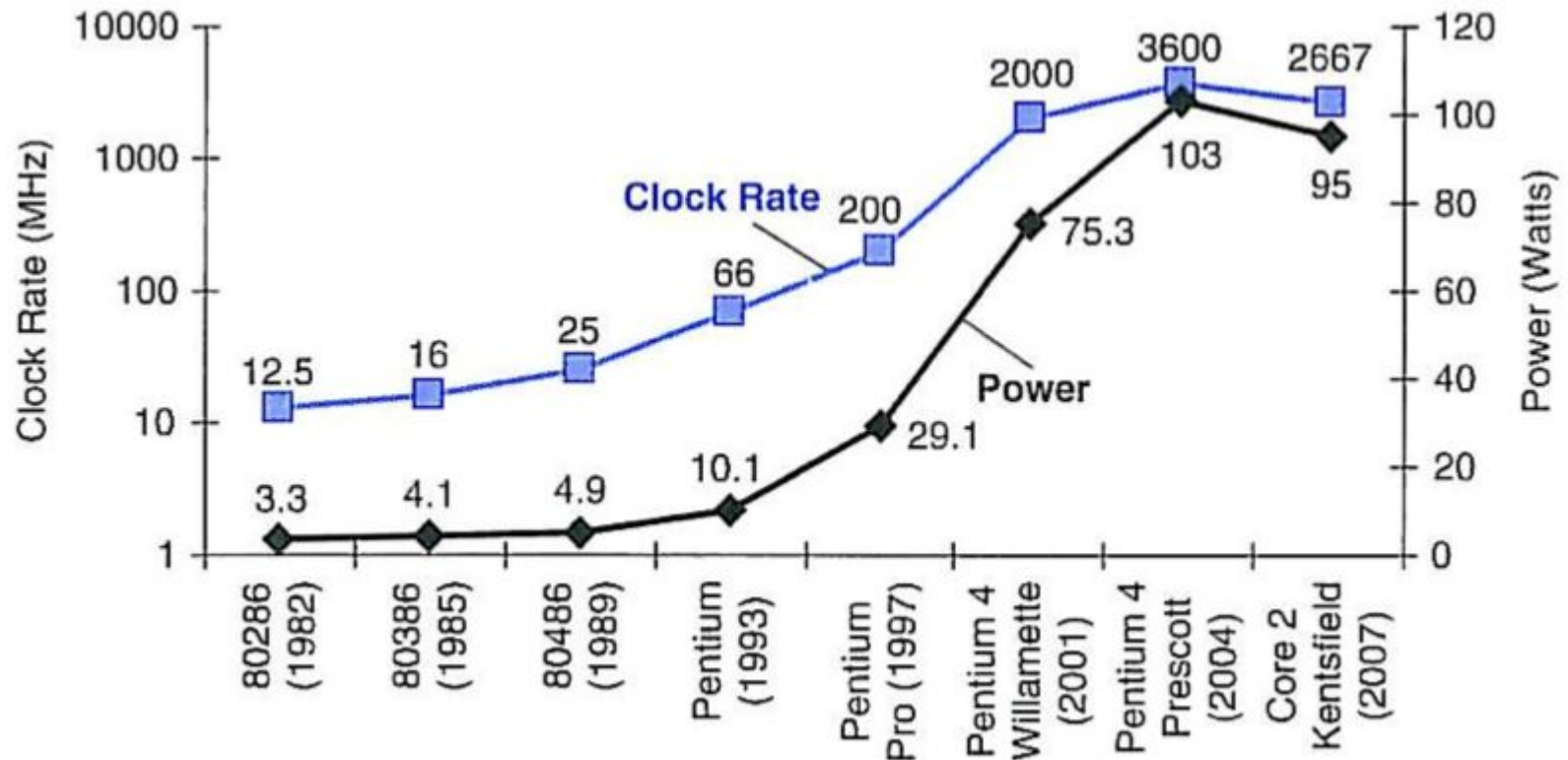


# Desempenho: resumo

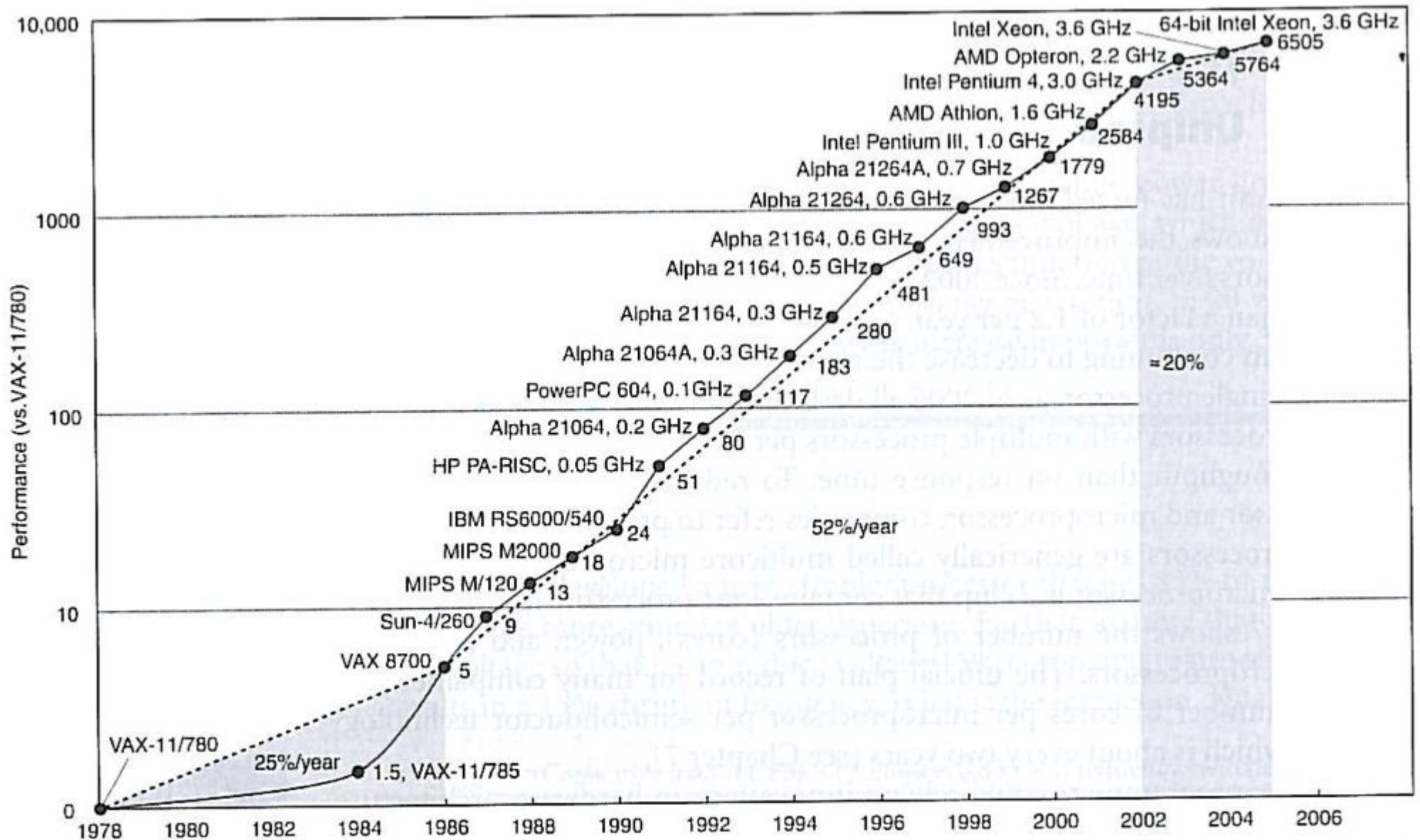
- Conjunto de instruções: o conjunto de instruções da CPU afetam a taxa do clock, o CPI e mesmo a quantidade de instruções que são usadas pelos programas
- Implementação interna do conjunto de instruções do processador afeta a taxa de clock e o CPI

# Avanços no Hardware

- Evolução dos processadores. Clock vs Energia



# Avanços no Hardware



# Avanços no Hardware

Product	AMD Opteron X4 (Barcelona)	Intel Nehalem	IBM Power 6	Sun Ultra SPARC T2 (Niagara 2)
Cores per chip	4	4	2	8
Clock rate	2.5 GHz	~ 2.5 GHz ?	4.7 GHz	1.4 GHz
Microprocessor power	120 W	~ 100 W ?	~ 100 W ?	94 W

# Avanços no Hardware

- Múltiplos cores: avanços na vazão em detrimento ao tempo de resposta
- No passado, programadores contavam com a Lei de Moore para garantir a velocidade de seus programas “A velocidade aumenta a cada 18 meses”
- Devido a barreira da potência, o tempo de resposta não é mais o referencial

# Avanços no Hardware

- Hoje, os programadores precisam compreender a arquitetura dos múltiplos processadores para aproveitar o ganho em processamento
- Contudo:
  - Não é trivial desenvolver códigos “paralelos”
  - Nem todo algoritmo tem uma implementação paralela eficiente
  - Os programas antigos, e mais comuns, não utilizam múltiplos processadores

# Exercicios

- 1.5, 1.6, 1.10, 1.12, 1.13, 1.14, 1.15, 1.16

→ (dois deles serão cobrados na prova!)