

Os principais componentes que afetam a performance:

Componentes da performance	Unidades de medida
Tempo de execução de um programa no processador	Segundos para o programa
Número de instruções	Instruções executadas por um programa
Ciclos de clock por instrução (CPI)	Número médio de ciclos de clock por instrução
Ciclo de clock	Segundos por ciclo de clock

$$\text{Tempo} = \frac{\text{Instruções}}{\text{Programa}} \times \frac{\text{Ciclos de clock}}{\text{Instrução}} \times \frac{\text{Segundos}}{\text{ciclo de clock}}$$

Como podemos medir o tempo de cada um desses parâmetros?

- Rodar um programa e medir o tempo de processador.
- Verificar a frequência nominal do processador.
- Número de instruções → simulador de uma arquitetura ou contar as instruções executadas (hardware).
- CPI → depende da organização de memória e processador (as instruções diferem em número de ciclos)

A CPI pode ser obtida por simulação detalhada de uma implementação ou por contadores de instruções implementados em hardware. Mas, podemos calcular a CPI em função do tipo de instrução empregada e por sua ocorrência em uma implementação:

$$\text{Número de ciclos de clock do processador} = \sum_{i=1}^n (\text{CPI}_i \times C_i)$$

Onde:

$C_i$  – é o número de instruções executadas de uma mesma classe  $i$ .

$\text{CPI}_i$  – é o número médio de ciclos por instrução para essa classe  $i$ .

$n$  – é o número total de classes de instruções.

Exemplo: Um projetista de compilador está tentando decidir entre duas seqüências de código para uma determinada máquina. Para tal, obteve com a equipe de hardware os seguintes dados:

Classe de instrução	CPI para esta classe de instrução
A	1
B	2
C	3

Considerando o código a ser gerado para uma particular declaração de uma linguagem de alto nível, o responsável pelo projeto do compilador está considerando duas possibilidades de seqüências de código, com as seguintes instruções:

Seqüência de código	Número de instrução para a classe		
	A	B	C
1	2	1	2
2	4	1	1

Qual das duas seqüências executa mais instruções? Qual a mais rápida? Qual a CPI para cada seqüência?

---

## 2.4 Escolha de programas para avaliar a Performance

Um usuário poderia comparar duas máquinas diferentes utilizando suas aplicações usuais como comparativo de desempenho.

Esta carga de programas que o usuário pode utilizar é conhecida como *workload*.

Como a maioria dos usuários utiliza diversas aplicações, eles devem utilizar uma *workload* apropriada para comparar o desempenho de máquinas diferentes.

**Benchmark** – conjunto de programas desenvolvidos para trabalhar na medida de performance. São baseados em aplicações reais utilizadas por grupos de usuários.

Grupos de engenheiros – o benchmark deve aplicações típicas de engenharia.

Grupos de engenheiros de software – o benchmark deve incluir compiladores e programas para processamento de documentos.

A utilização de aplicações reais limita o risco de burlar a execução do benchmark.

O uso de pequenos trechos de código como benchmark estimula a otimização tanto na arquitetura quanto no compilador para burlar o benchmark.

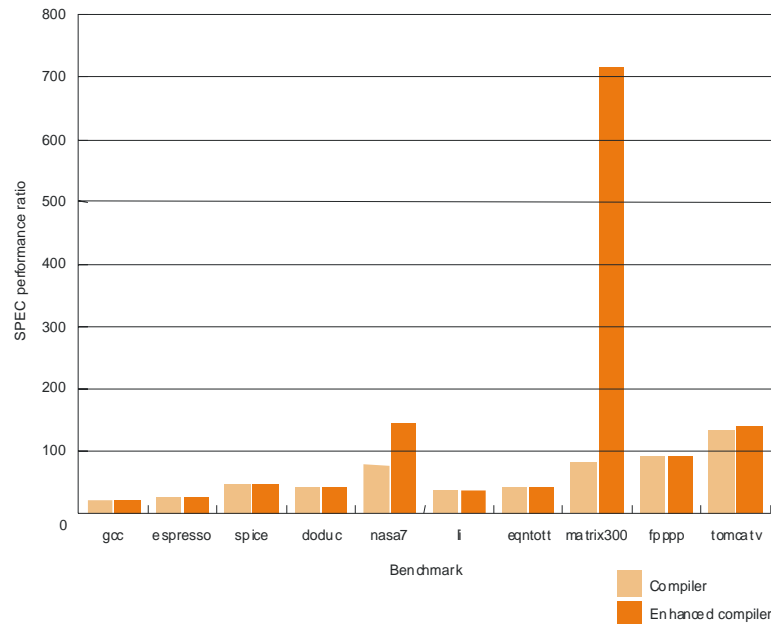
Software – o compilador pode reconhecer um trecho de código que pertence a m benchmark e gerar uma seqüência de instruções que sejam mais eficientes.

Hardware – o projetista da arquitetura pode fazer que um determinado conjunto de instruções seja executado mais rápido.

Casos:

Intel 95: (Standard Performance Evaluation Corporation) performance do Pentium usando um conjunto de benchmarks SPEC95 inteiro que usava um compilador da própria Intel e que não era disponível ao público. Código do benchmark estava errado e foi descoberto por concorrentes.

SPEC98: benchmark matrix300, utiliza uma série de multiplicação de matrizes. 99% do tempo de execução que era concentrada em um único trecho de código, dessa forma várias empresas desenvolveram compiladores especiais para apresentar melhoras desse benchmark.



Verificar o site: <http://www.spec.org/cpu2005/>

Informações sobre a performance do Pentium 4:

<http://www.digit-life.com/articles2/insidespeccpu/insidespeccpu2000-p4jan2003.html>

Porque as pessoas não utilizam programas reais para medir a performance?

Benchmarks pequenos podem ser facilmente compilados e simulados até mesmo através de cálculos manuais, além de fácil padronização.

Os benchmarks após a sua padronização devem ser publicados em relatórios técnicos

## 2.5 Comparação e documentação de performance

Ao invés de apresentar relatórios com a documentação sobre a Performance do processador muitos fabricantes preferem apresentar somente um número para representar a performance entre diferentes máquinas.

<b>Programas</b>	<b>Computador A</b>	<b>Computador B</b>
1 (em segundos)	1	10
2 (em segundos)	1000	100
Tempo total (em segundos)	1001	110

O que podemos concluir desse relatório?

Devemos comparar a performance em relação ao tempo de processamento, onde B é 9,1 mais rápido que A.

Se os programas são executados em máquinas diferentes o mesmo número de vezes, a média dos tempos de execução é a média aritmética:

$$MA = \frac{1}{n} \sum_{i=1}^n Tempo_i$$

Tempo<sub>i</sub> – tempo do i-ésimo programa do total de n pertencentes ao workload.

Se um programa rodar mais vezes que o outro podemos usar a média ponderada:

$$MP = \frac{w_1 P_1 + \dots + w_n P_n}{\sum_{i=1}^n w_i}$$

Podemos observar que quase sempre a melhoria de performance do processador são decorrentes de:

- Aumento da frequência do processador.
- Melhoria da organização interna do processador que diminui a CPI
- Modificações no projeto do compilador, que resultaram em uma qtde. menor de instruções executadas, com uma CPI média mais baixa (instruções mais simples).

## 2.6 Casos Reais: SPEC CPU 2000

### Documentação: Testbed configuration

The testbed is based on the Intel Pentium 4 3.06 GHz. All of the tests are carried out under the Microsoft Windows XP Pro SP1.

Platforms:

- Intel i850E (ASUS P4T533)
- Intel i845PE (Gigabyte GA-8PE667 Ultra)
- Intel E7205 (MSI GNB Max)
- SiS 655 (Gigabyte GA-SINXP1394)
- SiS R658 (ABIT SI7-G)

Each platform was coupled with a respective memory type - RDRAM for i850 and 658, DDR SDRAM for the rest. A bit later we will give you some more details on the memory configuration.

SPEC CPU2000 configuration: Intel C/C++/Fortran 7.0 compilers, Microsoft Visual.NET ([160103](#)) libraries. SSE2 optimization.

First of all, let me focus on the peculiarities of the systems tested and specify their possible memory configuration.

Processor		CPU bus bandwidth
Pentium 4, FSB 400 MHz		3.2 GB/s
Pentium 4, FSB 533 MHz		4.2 GB/s
Pentium 4, FSB 800 MHz		6.4 GB/s
Chipset	Bandwidth	
Intel i850E	4.2 GB/s (RDRAM PC4200)	
Intel i845PE	2.1 GB/s (DDR SDRAM PC2100)	
	2.7 GB/s (DDR SDRAM PC2700)	
Intel E7205	2.1 GB/s (DDR SDRAM PC2100)	
	4.2 GB/s (dual-channel DDR SDRAM PC2100)	
SiS655	from 2.1 GB/s (DDR SDRAM PC2100)	
	to 64 GB/s (dual-channel DDR SDRAM PC3200)	
SiS R658	4.2 GB/s (RDRAM PC4200)	
	4.8 GB/s (RDRAM PC4800)	

