

# Rendimiento y sus factores

# Rendimiento

- 1. m. Producto o utilidad que rinde o da alguien o algo.
- 2. m. Proporción entre el producto o el resultado obtenido y los medios utilizados.

© Real Academia Española (<http://dle.rae.es/?id=VwxnN6O>)

# Rendimiento

- ¿Cómo se puede definir el rendimiento de un sistema computacional?
- Una forma es mediante el **tiempo de respuesta**. Tiempo entre que llega y sale un proceso.
- Otra forma es usando el **throughput**. Cantidad de trabajo hecho en un tiempo dado.
- Una alternativa es contar el **número de instrucciones por segundo (IPS)** que se realizan.

# Rendimiento

- Una disminución del tiempo de respuesta implica un aumento de throughput.
- Pero, un aumento de throughput **no** siempre implica una disminución del tiempo de respuesta.

# Ejemplo

- Reemplazar CPUs viejas por nuevas disminuye el tiempo de respuesta y aumenta el throughput (cantidad de trabajo hecho).
- Aumentar el número de computadoras aumenta el throughput pero no disminuye el tiempo de respuesta.
- A menos que hubiera colas. En este caso, el aumento en el número de computadoras aumenta el throughput **y si** disminuye el tiempo de respuesta.

# Definición de rendimiento

- El rendimiento de una computadora se define en base al tiempo de ejecución de un programa en esa computadora.
- Para una computadora X su rendimiento es:

$$\text{Rendimiento}_x = \frac{1}{\text{Tiempo de ejecución}_x}$$

# Comparación de rendimiento

- Dos computadoras, X y Y, se pueden comparar mediante su rendimiento.
- X tiene mejor rendimiento que Y si:

$$\text{Rendimiento}_X > \text{Rendimiento}_Y$$

$$1 / \text{Tiempo de ejecución}_X > 1 / \text{Tiempo de ejecución}_Y$$

$$\text{Tiempo de ejecución}_X < \text{Tiempo de ejecución}_Y$$

- Es decir, si X es más rápida que Y, entonces se dice que X tiene mejor rendimiento que Y.

# Rendimiento relativo

- “X es n veces más rápida que Y” significa:

$$\frac{\text{Rendimiento X}}{\text{Rendimiento Y}} = n > 1$$

- Alternativamente:

$$\frac{\text{Tiempo de ejecución Y}}{\text{Tiempo de ejecución X}} = n > 1$$

# Ejemplo

- La computadora A corre un programa en 10 segundos y la computadora B en 15 segundos.

$$\begin{aligned} \text{RendimientoA} / \text{RendimientoB} &= \\ \text{Tiempo de ejecuciónB} / \text{Tiempo de ejecuciónA} &= \\ 15 / 10 &= \\ 1.5 & \end{aligned}$$

- Conclusión: A es 1.5 veces más rápida que B.

# Tiempo

1. Tiempo de respuesta. Tiempo total para realizar una tarea. Incluye:

- Tiempo de espera en cola.
- Accesos a disco.
- Accesos a memoria.
- Actividades de entrada y salida (I/O).
- *Overhead* del sistema operativo.

Tiempo que el sistema operativo dedica a sus tareas propias y no al proceso que está ejecutando. Tiempo de carga, scheduling, recolección de basura, etc.

# Tiempo

2. Tiempo de CPU. Tiempo que la CPU dedica a la tarea. No incluye tiempo dedicado a correr otras tareas.

Se puede dividir en:

- Tiempo de CPU de usuario. Tiempo que la CPU dedica al código de la tarea.
- Tiempo de CPU de sistema. Tiempo que la CPU dedica al sistema operativo cuando éste realiza actividades relacionadas con la tarea. Por ejemplo, tiempo para cargar la tarea en memoria.

# Tiempo

- En conclusión, se distinguen dos tiempos:
  1. Tiempo de respuesta en un sistema sin carga (sin otros usuarios).
  2. Tiempo de CPU de usuario.

# Reloj

El reloj del sistema determina cuando los eventos ocurren en el hardware.

1. Periodo de reloj. Tiempo en que ocurre un ciclo (pulso) de reloj. Se mide en segundos. Por ejemplo 0.25 nanosegundos.
2. Velocidad de reloj. Es el inverso del periodo. Se mide en ciclos por segundo. Por ejemplo, 4GHz son  $4 \times 10^9$  ciclos por segundo.

# Rendimiento de la CPU

- El tiempo de ejecución de CPU de un programa:

Tiempo CPU = Ciclos de reloj para el programa x Periodo del reloj

- Alternativamente

Tiempo CPU = Ciclos de reloj para el programa / Velocidad del reloj

# Rendimiento de la CPU

- El tiempo de CPU de un programa puede mejorar (disminuir) si:
  - El programa corre más rápido (usa menos ciclos de reloj).
  - El periodo de reloj de la CPU disminuye y por lo tanto, aumenta la velocidad del reloj.

# Ejemplo

- Un programa tarda 10 segundos en correr en una computadora A que tiene una velocidad de reloj de 4 GHZ.
- Se desea construir una computadora B que corra el mismo programa en 6 segundos.
- Por problemas de diseño, B va a necesitar 1.2 veces mas ciclos de reloj que A para hacer la misma tarea.
- ¿Cuál debe ser la velocidad de reloj de B para cumplir la meta?

# Ejemplo

- Encontrar el número de ciclos que usa el programa en A.

$$\text{Tiempo CPU}_A = \text{Ciclos reloj}_A / \text{Velocidad reloj}_A$$

$$10 \text{ segundos} = \text{Ciclos reloj}_A / (4 \times 10^9 \text{ ciclos / segundo})$$

$$\text{Ciclos reloj}_A = 10 \text{ segundos} \times (4 \times 10^9 \text{ ciclos / segundo})$$

$$= 40 \times 10^9 \text{ ciclos.}$$

- Para correr en A el programa usa  $40 \times 10^9$  ciclos de reloj.

# Ejemplo

- Encontrar la velocidad de B, sabiendo que B va a necesitar  $1.2 \times 40 \times 10^9$  ciclos para correr el programa.

$$\text{Tiempo CPU}_B = \text{Ciclos reloj}_B / \text{Velocidad reloj}_B$$

$$6 \text{ segundos} = (1.2 \times 40 \times 10^9 \text{ ciclos}) / \text{Velocidad reloj}_B$$

$$\text{Velocidad reloj}_B = (1.2 \times 40 \times 10^9 \text{ ciclos}) / 6 \text{ segundos}$$

$$= 8 \times 10^9 \text{ ciclos / segundo}$$

$$= 8 \text{ GHz}$$

- B debe tener el doble de velocidad que A para ejecutar el programa en 6 segundos.

# Ciclos de reloj

- En la ecuación

Tiempo CPU = Ciclos de reloj / Velocidad del reloj

es tedioso contar el número de ciclos de reloj que usa un programa.

- Otra opción:

Ciclos de reloj = Número de instrucciones x CPI

CPI = Promedio de ciclos de reloj por instrucción.

# CPI

- El CPI es útil para comparar dos implementaciones de un conjunto de instrucciones.
- Para esto, se comparan los tiempos de ejecución de un mismo programa en las dos implementaciones.

# Ejemplo

- La computadora A tiene un periodo de reloj de 250 ps (pico segundos) y un CPI = 2 para cierto programa.
- La computadora B tiene un periodo de reloj de 500 ps y un CPI = 1.2 para el mismo programa.
- ¿Qué computadora es más rápida y en cuanto?

# Ejemplo

- Se usan la ecuaciones:
  - Tiempo CPU = Ciclos de reloj para el programa x Periodo del reloj
  - Ciclos de reloj = Número de instrucciones x CPI
  - Tiempo de CPU<sub>A</sub> / Tiempo de CPU<sub>B</sub> = n

# Ejemplo

- Tiempo de CPU para A:

$$\begin{aligned}\text{Tiempo CPU}_A &= \text{Ciclos de reloj}_A \times \text{Periodo del reloj}_A \\ &= \text{Ciclos de reloj}_A \times 250 \text{ ps}\end{aligned}$$

- Ciclos de reloj del programa en la CPU A

$$\begin{aligned}\text{Ciclos de reloj}_A &= \text{Número de instrucciones}_A \times \text{CPI}_A \\ &= 1 \times 2\end{aligned}$$

- Al final

$$\begin{aligned}\text{Tiempo CPU}_A &= 1 \times 2 \times 250 \text{ ps} \\ &= 1 \times 500 \text{ ps}\end{aligned}$$

# Ejemplo

- Tiempo de CPU para B:

$$\begin{aligned}\text{Tiempo CPU}_B &= \text{Ciclos de reloj}_B \times \text{Periodo del reloj}_B \\ &= \text{Ciclos de reloj}_B \times 500 \text{ ps}\end{aligned}$$

- Ciclos de reloj del programa en la CPU B

$$\begin{aligned}\text{Ciclos de reloj}_B &= \text{Número de instrucciones}_B \times \text{CPI}_B \\ &= 1 \times 1.2\end{aligned}$$

- Al final

$$\begin{aligned}\text{Tiempo CPU}_B &= 1 \times 1.2 \times 500 \text{ ps} \\ &= 1 \times 600 \text{ ps}\end{aligned}$$

# Conclusión del ejemplo

- Tiempo  $\text{CPU}_A = 1 \times 500 \text{ ps.}$
- Tiempo  $\text{CPU}_B = 1 \times 600 \text{ ps.}$
- A es más rápida que B.

$$\begin{aligned} & \text{Tiempo de CPU}_B / \text{Tiempo de CPU}_A \\ & 1 \times 600 \text{ ps} / 1 \times 500 \text{ ps} = 1.2 \end{aligned}$$

- A es más rápida que B 1.2 veces.

# Rendimiento de la CPU

- La ecuación de rendimiento de la CPU puede reescribirse como:

Tiempo CPU = Número de instrucciones x CPI x Periodo del reloj

- Alternativamente

Tiempo CPU = (Número de instrucciones x CPI) / Velocidad del reloj

- No es necesario contar el número de ciclos de reloj del programa.

# Factores del rendimiento

- Los factores son los elementos de la ecuación de rendimiento de la CPU.
  1. Número de instrucciones del programa.
  2. CPI (Número de ciclos por instrucción).
  3. Periodo de reloj / velocidad del reloj.
- En vez del número de instrucciones y el CPI se puede usar el número de ciclos del programa.

# Factores del rendimiento

- ¿Cómo se obtiene cada elemento de la ecuación de rendimiento?
- El tiempo de CPU se obtiene corriendo el programa.
- La velocidad, y por lo tanto el periodo del reloj, es parte de la documentación de la computadora.
- El número de instrucciones y el CPI pueden ser más complicados.

# Factores del rendimiento

- El número de instrucciones se puede obtener:
  - Contando las instrucciones (sin seudos) del programa.
  - Usando un simulador del ISA.
  - Usando un programa *profiler*.
  - Usando contadores de hardware (si están disponibles).
- El CPI se puede obtener:
  - Usando un simulador de la implementación.
  - Usando contadores de hardware.

# Consideraciones

- El número de instrucciones depende del conjunto de instrucciones (el ISA) y no de la implementación.
- El CPI depende de la implementación porque depende de:
  - La estructura de la memoria.
  - La estructura del procesador.
  - Las instrucciones usadas en el programa.
- Dos implementaciones de MIPS pueden tener distintos CPI.

# Número de ciclos

- El número de ciclos se puede aproximar usando:

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

- donde
  - $C_i$  es el número de instrucciones de clase  $i$  ejecutadas.
  - $\text{CPI}_i$  es el promedio del número de ciclos por instrucción para la clase  $i$ .
  - $n$  es el número de clases de instrucciones.

# Ejemplo

- Hay 3 clases de instrucciones: A, B y C, cada una con su propio CPI.

	CPI por clase		
	A	B	C
CPI	1	2	3

# Ejemplo

- Hay dos secuencias de código,  $S_1$  y  $S_2$ , que se quieren comparar.
- Cada secuencia requiere el siguiente número de instrucciones de cada clase:

Secuencia de código	Número de instrucciones de cada clase		
	A	B	C
$S_1$	2	1	2
$S_2$	4	1	1

# Ejemplo

1. ¿Qué secuencia ejecuta más instrucciones?
2. ¿Qué secuencia es más rápida?
3. ¿Cuál es el CPI de cada secuencia?

# Ejemplo

1. ¿Qué secuencia ejecuta más instrucciones?

$S_1$  ejecuta  $2 + 1 + 2 = 5$  instrucciones.

$S_2$  ejecuta  $4 + 1 + 1 = 6$  instrucciones.

- $S_1$  ejecuta menos instrucciones que  $S_2$ .

# Ejemplo

2. ¿Qué secuencia es más rápida?  
es decir, ¿Qué secuencia ejecuta menos ciclos?

- Se usa la ecuación:

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

Ciclos de  $S_1 = (2 \times 1) + (1 \times 2) + (2 \times 3) = 10$  ciclos

Ciclos de  $S_2 = (4 \times 1) + (1 \times 2) + (1 \times 3) = 9$  ciclos

- $S_2$  es más rápida que  $S_1$ .

# Ejemplo

3. ¿Cuál es el CPI de cada secuencia?

- Se usa la ecuación:

$$\text{CPI} = \text{Número de ciclos} / \text{Número de instrucciones.}$$

- Para  $S_1$   $\text{CPI} = 10 / 5 = 2$
- Para  $S_2$   $\text{CPI} = 9 / 6 = 1.5$

# MIPS

- MIPS – Millones de instrucciones por segundo.
- Es una medida alternativa del rendimiento.
- Problemas:
  - a) Los MIPS no siempre son relevantes, p.e., en servidores de web, de correo o de archivos.
  - b) Un programa puede tener varios MIPS dependiendo del compilador utilizado.
  - c) Una computadora puede realizar más MIPS que otra y al mismo tiempo ser más lenta.

# MIPS

- Para calcular las IPS (instrucciones por segundo):

$$IPS = \frac{I}{T}$$

- Para calcular los MIPS (millones de IPS):

$$MIPS = \frac{I}{T \times 10^6} = \frac{IPS}{1000000}$$

- Ecuación del tiempo:

$$T = \frac{I \times CPI}{V}$$

- Se sustituye la ecuación del tiempo en la de MIPS.

# MIPS

$$MIPS = \frac{I}{\frac{I \times CPI \times 10^6}{V}} = \frac{V}{CPI \times 10^6}$$

- Un programa puede tener varios CPIs, según el compilador usado, y no tener un solo MIPS.
- Una computadora puede tener un MIPS mayor que otra y al mismo tiempo ser más lenta.

# Ejemplo

Measurement	Computer A	Computer B
Instruction count	10 billion	8 billion
Clock rate	4 GHz	4 GHz
CPI	1.0	1.1

- ¿Qué computadora tiene un MIPS mayor?
- ¿Qué computadora es más rápida?

# Ejemplo

- MIPS = (Velocidad reloj / CPI x  $10^6$ )

- Computadora A

$$\text{MIPS} = 4 \times 10^9 / (1.0 \times 10^6) = 4 \times 10^3$$

- Computadora B

$$\text{MIPS} = 4 \times 10^9 / (1.1 \times 10^6) = 3.6 \times 10^3$$

- Conclusión: A tiene MIPS mayor.

# Tiempo de ejecución

- $T = I \times \text{CPI} / V$
- Computadora A  
$$T = 10 \times 10^9 \times 1.0 / 4 \times 10^9 = 2.5 \text{ segundos}$$
- Computadora B  
$$T = 8 \times 10^9 \times 1.1 / 4 \times 10^9 = 2.2 \text{ segundos}$$
- Conclusión: B es más rápida.

# La mejor medida

- La mejor medida de rendimiento es el tiempo.

Execution time is the only valid and unimpeachable measure of performance. Many other metrics have been proposed and found wanting. Sometimes these metrics are flawed from the start by not reflecting execution time; other times a metric that is valid in a limited context is extended and used beyond that context or without the additional clarification needed to make it valid.

- Fuente: COD, p. 52

# Resumen

- Los factores básicos del rendimiento de la CPU y sus medidas.

Componente	Unidad de medida
Tiempo de ejecución de CPU para un programa.	Segundos.
Número de instrucciones.	Instrucciones ejecutadas por el programa.
Ciclos de reloj por instrucción (CPI).	Número de ciclos de reloj en promedio por instrucción.
Periodo de reloj.	Segundos por ciclo de reloj.

# Resumen

- Sin olvidarse de:
  - Número de ciclos de reloj del programa = número de instrucciones x CPI.
  - Velocidad del reloj =  $1 / \text{periodo de reloj}$ .

# Resumen

- Al comparar dos computadoras no basta fijarse en una sola medida, hay que ver las 3 al mismo tiempo:
  - Número de instrucciones.
  - CPI.
  - Periodo de reloj.
- Las 3 medidas juntas forman el tiempo de ejecución de CPU del programa.

# Resumen

- Las partes de un programa afectan su tiempo de ejecución.

Componente	¿Qué afectan?
Algoritmo.	Número de instrucciones. Tal vez el CPI.
Lenguaje de programación.	Número de instrucciones. CPI.
Compilador	Número de instrucciones. CPI.
ISA (conjunto de instrucciones)	Número de instrucciones. CPI. Periodo de reloj.