

U.B.A. - Facultad de Ingeniería

66.20/86.37 Organización de Computadoras

Desempeño

Práctica

1<sup>er</sup> cuatrimestre 2020

# Desempeño

Medir la performance en forma precisa y correcta para comparar distintas computadoras es crítica para los compradores y por lo tanto para los diseñadores.

Para maximizar la performance vamos a querer minimizar el tiempo de respuesta o el tiempo de ejecución de una determinada tarea. Con lo cual, podemos relacionar la performance y el tiempo de ejecución para una computadora X.

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

# Desempeño

Si la performance de X es mayor que la performance de Y, tenemos que:

$$\begin{aligned} \text{Performance}_x &> \text{Performance}_y \\ \frac{1}{\text{Tiempo de ejecución}_x} &> \frac{1}{\text{Tiempo de ejecución}_y} \\ \text{Tiempo de ejecución}_y &> \text{Tiempo de ejecución}_x \end{aligned}$$

El tiempo de ejecución de Y es mayor que el de X, es decir, X es más rápida que Y.

# Benchmarks

- ▶ kernels, partes pequeñas y significativas de una aplicación real.
- ▶ programas de juguete, son programas de no más de 100 líneas, como el quicksort.
- ▶ benchmarks sintéticos, son programas inventados que aprovechan ciertas características de una determinada computadora, como Dhrystone.

# Ley de Amdahl

La ley de Amdahl define el speedup que puede ser obtenido al utilizar una mejora particular.

El speedup está determinado por la relación:

$$\text{speedup} = \frac{\text{Performance de toda la tarea utilizando la mejora cuando es posible}}{\text{Performance de toda la tarea sin ninguna mejora}}$$

O de otro modo,

$$\text{speedup} = \frac{\text{Tiempo de ejecución de toda la tarea sin utilizar la mejora}}{\text{Tiempo de ejecución de toda la tarea utilizando la mejora cuando es posible}}$$

# Ley de Amdahl

La ley de Amdahl entonces nos brinda una forma rápida de encontrar el speedup de una mejora que depende de dos factores:

1. La fracción de tiempo del tiempo original que se puede aplicar una mejora.
2. La mejora obtenida al aplicar el nuevo modo, esto es, que tan rápido correrá el sistema si se aplica la mejora.

$$\text{Tiempo de ejecución}_{\text{nuevo}} = \text{Tiempo de ejecución}_{\text{viejo}} \times \left( (1 - \text{Fraccion}_{\text{mejorable}}) + \frac{\text{Fraccion}_{\text{mejorable}}}{\text{speedup}_{\text{local}}} \right)$$

# Ecuación de desempeño del CPU

CPU time = Instruction count  $\times$  CPI  $\times$  Clock cycle time

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

Componente de performance	Unidad de medida
Tiempo de ejecución de CPU para un programa	Segundos por programa
Instruction count (Cantidad de instrucciones)	Instrucciones ejecutadas por el programa
Ciclos de reloj por instrucción (CPI)	Promedio de ciclos de reloj por instrucción
Tiempo de ciclo de reloj (Clock cycle time)	Segundos por ciclo de reloj

$$\text{Time} = \frac{\text{Seconds}}{\text{Program}} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Clock Cycles}}{\text{Instruction}} \times \frac{\text{Seconds}}{\text{Clock Cycles}}$$

# Medición de desempeño

- ▶ CPU execution time o CPU time. El tiempo total que toma la CPU en computar una tarea específica.
- ▶ user CPU time. El tiempo de CPU que toma el programa en sí.
- ▶ system CPU time. El tiempo de CPU que toma el sistema operativo realizando tareas para el programa.

Una alternativa a medir tiempo es MIPS (millones de instrucciones por segundo). Para un dado programa, MIPS se calcula:

$$\text{MIPS} = \frac{\text{Intruction count}}{\text{Execution time} \times 10^6}$$



## Ejercicio 1.2 CAAQA 3ed

In this exercise, assume that we are considering enhancing a machine by adding vector hardware to it. When a computation is run in vector mode on the vector hardware, it is 10 times faster than the normal mode of execution. We call the percentage of time that could be spent using vector mode the percentage of vectorization. Vectors are discussed in Appendix G, but you don't need to know anything about how they work to answer this question!

## Ejercicio 1.2 CAAQA 3ed

- a. Draw a graph that plots the speedup as a percentage of the computation performed in vector mode. Label the y-axis “Net speedup” and label the x-axis “Percent vectorization.”
- b. What percentage of vectorization is needed to achieve a speedup of 2?
- c. What percentage of the computation run time is spent in vector mode if a speedup of 2 is achieved?

## Ejercicio 1.2 CAAQA 3ed

- d. What percentage of vectorization is needed to achieve one-half the maximum speedup attainable from using vector mode?
- e. Suppose you have measured the percentage of vectorization for programs to be 70%. The hardware design group says they can double the speed of the vector hardware with a significant additional engineering investment. You wonder whether the compiler crew could increase the use of vector mode as another approach to increasing performance. How much of an increase in the percentage of vectorization (relative to current usage) would you need to obtain the same performance gain as doubling vector hardware speed? Which investment would you recommend?

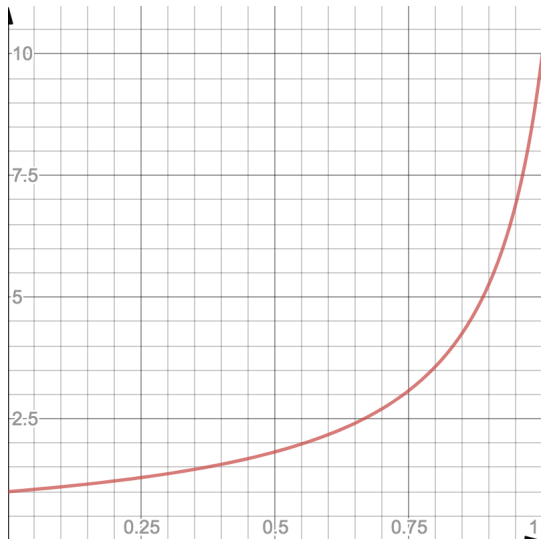
## Ejercicio 1.2 CAAQA 3ed - Resolución

a.

Partiendo de la Ley de Amdahl, así como los datos del enunciado, tenemos que  $SU_g = \frac{1}{(1-f) + \frac{f}{10}} = \frac{1}{1-0,9 \times f}$ . Sabemos también que  $f \in [0, 1]$ . Podemos calcular el valor de  $SU$  en los extremos:

- ▶ Si no es posible aplicar la mejora,  $f = 0 \implies SU_g = 1$
- ▶ Si la mejora es aplicable a todo el tiempo de ejecución,  $f = 1 \implies SU_g = SU_L = 10$

## Ejercicio 1.2 CAAQA 3ed - Resolución



## Ejercicio 1.2 CAAQA 3ed - Resolución

b.

¿Qué porcentaje de vectorización es necesaria para alcanzar un  $SU$  global de 2?

Utilizando la ley de Amdahl podemos despejar  $f$

$$2 = \frac{1}{1 - 0,9 \times f}$$
$$f = \frac{5}{9}$$

## Ejercicio 1.2 CAAQA 3ed - Resolución

c.

Partiendo de la definición de  $SU$ , tenemos que

$$SU_g = \frac{T_O}{T_N} = 2 \implies f = \frac{5}{9}$$

$$SU_L = \frac{T_m}{T'_m} = 10, \quad \text{siendo} \quad T_m = T_O \times f = \frac{5}{9} T_O$$

Es importante notar que  $T'_{nm} = T_{nm} = \frac{4}{9} \times T_O$ , ya que el tiempo en el que no se aplica ninguna optimización no cambia.

## Ejercicio 1.2 CAAQA 3ed - Resolución

c.

El ejercicio nos pide calcular  $f' = \frac{T'_m}{T_N}$ . Se puede determinar que

$$T'_m = \frac{1}{10} \times T_m$$

$$T_N = T'_m + T'_{nm}$$

$$T_{nm} = T'_{nm}$$

$$f' = \frac{\frac{5}{90}}{\frac{5}{90} + \frac{4}{9}}$$



## Ejercicio 1.2 CAAQA 3ed - Resolución

d.

A partir de la Ley de Amdahl y el análisis realizado en el primer punto sabemos que  $SU_{max} = 10$ , por lo que solo resta resolver

$$\frac{1}{5} = 1 - 0,9 \times f$$

$$f = \frac{8}{9}$$

## Ejercicio 1.2 CAAQA 3ed - Resolución

e.

Para igualar el speedup por hardware de 20 aplicado al 70% de vectorización tenemos que

$$\frac{1}{1-f+\frac{f}{10}} = \frac{1}{1-70\%+\frac{70\%}{20}}$$

Incrementar el porcentaje de vectorización  $f$  un 74%. Con lo cual mejorar el compilador el compilador un 4% va a ser más simple y más económico que mejorar el hardware por un factor de 2.

## Ejercicio 1.16 CAAQA 3ed

Three enhancements with the following speedups are proposed for a new architecture:

- ▶  $\text{Speedup1} = 30$
- ▶  $\text{Speedup2} = 20$
- ▶  $\text{Speedup3} = 15$

Only one enhancement is usable at a time.

## Ejercicio 1.16 CAAQA 3ed

- a. If enhancements 1 and 2 are each usable for 25% of the time, what fraction of the time must enhancement 3 be used to achieve an overall speedup of 10?
- b. Assume the enhancements can be used 25%, 35%, and 10% of the time for enhancements 1, 2, and 3, respectively. For what fraction of the reduced execution time is no enhancement in use?
- c. Assume, for some benchmark, the possible fraction of use is 15% for each of enhancements 1 and 2 and 70% for enhancement 3. We want to maximize performance. If only one enhancement can be implemented, which should it be? If two enhancements can be implemented, which should be chosen?

## Ejercicio 1.16 CAAQA 3ed - Resolución

Podemos generalizar la ley de Amdahl cuando tenemos varias mejoras aplicadas en curso

$$\text{speedup} = \left[ 1 - \sum_i FE_i + \sum_i \frac{FE_i}{SE_i} \right]^{-1}$$

Donde  $FE_i$  es la fracción del tiempo donde la mejora  $i$  se puede utilizar.  $SE_i$  es el respectivo speedup local para la mejora  $i$

## Ejercicio 1.16 CAAQA 3ed - Resolución

a.

Con tres mejoras tenemos:

$$\text{speedup} = 1 - (FE_1 + FE_2 + FE_3) + \frac{FE_1}{SE_1} + \frac{FE_2}{SE_2} + \frac{FE_3}{SE_3}$$

Reemplazando:

$$10 = 1 - (0.25 + 0.25 + FE_3) + \frac{0.25}{30} + \frac{0.25}{20} + \frac{FE_3}{15}$$

Resolviendo:

$$FE_3 = 0.45$$

## Ejercicio 1.16 CAAQA 3ed - Resolución

b.

- ▶  $T_e$ : Tiempo de ejecución mejorado con mejoras aplicadas
- ▶  $TNE_e$ : Tiempo de ejecución mejorado sin mejoras aplicadas
- ▶  $T_{original}$ : Tiempo de ejecución sin mejoras
- ▶  $FNE_{original}$ : Fracción del tiempo que no puede ser mejorado

## Ejercicio 1.16 CAAQA 3ed - Resolución

b.

Dado que el tiempo consumido por el código que no puede ser mejorado es el mismo cuando se aplican las mejoras que cuando no, tenemos:

$$\frac{TNE_e}{T_e} = \frac{FNE_{original} \times T_{original}}{T_{original}/\text{speedup}}$$

Si reemplazamos por la expresión equivalente de  $FNE_{original}$  y el speedup tenemos:

$$\frac{FNE_{original} \times T_{original}}{T_{original}/\text{speedup}} = \frac{1 - \sum_i FE_i}{1 - \sum_i FE_i + \sum_i \frac{FE_i}{SE_i}}$$



## Ejercicio 1.16 CAAQA 3ed

b.

$$FNE_e = \frac{1 - (0.25 + 0.35 + 0.1)}{1 - (0.25 + 0.35 + 0.1) + \left(\frac{0.25}{30} + \frac{0.35}{20} + \frac{0.10}{15}\right)} = \frac{0.3}{0.3325} = 90\%$$

## Ejercicio 1.16 CAAQA 3ed

c.

Calculamos el speedup cuando sólo una mejora puede estar en uso

$$\text{speedup}_1 = \left(1 - 0.15 + \frac{0.15}{30}\right)^{-1} = 1.17$$

$$\text{speedup}_2 = \left(1 - 0.15 + \frac{0.15}{20}\right)^{-1} = 1.17$$

$$\text{speedup}_3 = \left(1 - 0.7 + \frac{0.7}{15}\right)^{-1} = 2.88$$

## Ejercicio 1.16 CAAQA 3ed

c.

$$\text{speedup}_12 = \left[ 1 - (0.15 + 0.15) + \frac{0.15}{30} + \frac{0.15}{20} \right]^{-1} = 1.40$$

$$\text{speedup}_13 = \left[ 1 - (0.15 + 0.7) + \frac{0.15}{30} + \frac{0.7}{15} \right]^{-1} = 4.96$$

$$\text{speedup}_23 = \left[ 1 - (0.15 + 0.7) + \frac{0.15}{20} + \frac{0.7}{15} \right]^{-1} = 4.90$$

## Ejercicio 1.17 CAAQA 3ed

Your company has a benchmark that is considered representative of your typical applications. An embedded processor under consideration to support your task does not have a floating-point unit and must emulate each floating-point instruction by a sequence of integer instructions. This processor is rated at 120 MIPS on the benchmark. A third-party vendor offers a compatible coprocessor to boost performance. That coprocessor executes each floating-point instruction in hardware (i.e., no emulation is necessary). The processor/coprocessor combination rates 80 MIPS on the same benchmark.

## Ejercicio 1.17 CAAQA 3ed

The following symbols are used to answer parts (a)–(e) of this exercise:

- ▶ I: Number of integer instructions executed on the benchmark.
- ▶ F: Number of floating-point instructions executed on the benchmark. Y—Number of integer instructions to emulate one floating-point instruction. W—Time to execute the benchmark on the processor alone.
- ▶ B: Time to execute the benchmark on the processor/coprocessor combination.

## Ejercicio 1.17 CAAQA 3ed

- a. Write an equation for the MIPS rating of each configuration using the symbols above.
- b. For the configuration without the coprocessor, we measure that  $F = 8 \times 10^6$ ,  $Y = 50$ , and  $W = 4$  seconds. Find  $I$ .
- c. What is the value of  $B$ ?
- d. What is the MFLOPS rating of the system with the coprocessor?
- e. Your colleague wants to purchase the coprocessor even though the MIPS rating for the configuration using the coprocessor is less than that of the processor alone. Is your colleague's evaluation correct? Defend your answer.

## Ejercicio 1.17 CAAQA 3ed - Resolución

a.

$$MIPS_{proc} = 120 \times 10^6 = \frac{I + YF}{W}$$

$$MIPS_{proc/co} = 80 \times 10^6 = \frac{I + F}{B}$$

## Ejercicio 1.17 CAAQA 3ed - Resolución

b.

$$\begin{aligned} I &= 120 \times 10^6 \times W - FY \\ &= (120 \times 10^6)(4) - (8 \times 10^6)(50) \\ &= 80 \times 10^6 \text{ instrucciones} \end{aligned}$$



## Ejercicio 1.17 CAAQA 3ed - Resolución

c.

$$B = \frac{80 \times 10^6 + 8 \times 10^6}{80 \times 10^6} = 1.1 \text{sec}$$

## Ejercicio 1.17 CAAQA 3ed - Resolución

e.

$$\begin{aligned} MFLOPS_{proc/co} &= \frac{F}{B - \text{T tiempo para las } I \text{ enteras}} \\ &= \frac{F}{B - I / MIPS_{proc/co}} \\ &= \frac{8 \times 10^6}{1.1 - 80 \times 10^6 / 80 \times 10^6} \\ &= 80 MFLOPS \end{aligned}$$