

بنيان الحواسيب

محاضرة 3

تقييم أداء أجهزة الكمبيوتر

د. فادي متوج

تقييم أداء أجهزة الكمبيوتر

- عند محاولة الاختيار من بين أجهزة كمبيوتر مختلفة، يعد الأداء سمة مهمة. لذلك فإن القياس الدقيق ومقارنة أجهزة الكمبيوتر المختلفة أمر بالغ الأهمية للمشتريين وبالتالي للمصممين.
- يرغب البائعون في أن ترى أجهزة الكمبيوتر الخاصة بهم في أفضل صورة ممكنة، سواء كان هذه الصورة تعكس بدقة احتياجات المشتري أم لا.
- وبالتالي فإن فهم أفضل السبل لقياس الأداء والقيود المفروضة على قياسات الأداء أمر مهم في اختيار جهاز الكمبيوتر.
- سوف ندرس في هذه المحاضرة الطرق المختلفة التي يمكن من خلالها تقييم الأداء ؛ بعد ذلك، نصف مقاييس قياس الأداء من وجهة نظر كل من مستخدم الكمبيوتر والمصمم. ننظر أيضًا في كيفية ارتباط هذه المقاييس ونقدم معادلة أداء المعالج الكلاسيكية، والتي سنستخدمها في هذا المقرر.

تحديد الأداء

- عندما نقول أن أحد أجهزة الكمبيوتر لديه أداء أفضل من الآخر، ماذا نعني؟ على الرغم من أن هذا السؤال قد يبدو بسيطاً، إلا أن التشابه مع طائرات الركاب يوضح مدى دقة مسألة الأداء.
- يوضح الشكل بعض طائرات الركاب النموذجية، جنباً إلى جنب مع سرعتها ونطاقها وسعتها.
- إذا أردنا معرفة أي من الطائرات في هذا الجدول لديها أفضل أداء، فسنحتاج أولاً إلى تحديد الأداء المقصود. على سبيل المثال، بالنظر إلى مقاييس الأداء المختلفة، نرى أن الطائرة ذات أعلى سرعة هي كونكورد، والطائرة ذات المدى الأطول هي DC-8، والطائرة ذات السعة الأكبر هي 747.

Airplane	Passenger capacity	Cruising range (miles)	Cruising speed (m.p.h.)	Passenger throughput (passengers × m.p.h.)
Boeing 777	375	4630	610	228,750
Boeing 747	470	4150	610	286,700
BAC/Sud Concorde	132	4000	1350	178,200
Douglas DC-8-50	146	8720	544	79,424

تحديد الأداء

- وبالمثل، يمكننا تحديد أداء الكمبيوتر بعدة طرق مختلفة.
- إذا كنا نشغل برنامجًا على جهاز كمبيوتر مختلفين، فيمكننا القول إن الكمبيوتر الأسرع هو الذي ينجز المهمة أولاً.
- في حالة مركز يحتوي على العديد من الخوادم التي تقوم بتنفيذ المهام المرسله من قبل العديد من المستخدمين، فيمكن القول إن الكمبيوتر الأسرع هو الذي أكمل معظم المهام خلال اليوم.
- بصفتنا مستخدمًا فرديين للكمبيوتر، فنحن مهتمون بتقليل **زمن الاستجابة** - الزمن بين بدء المهمة وإكمالها - يشار إليه أيضًا باسم **زمن التنفيذ Execution Time**.
- غالبًا ما يهتم مديرو مراكز البيانات بزيادة الإنتاجية - إجمالي حجم العمل المنجز في وقت معين. وبالتالي، في معظم الحالات، سنحتاج إلى مقاييس أداء مختلفة بالإضافة إلى مجموعات مختلفة من التطبيقات لقياس أداء أجهزة الكمبيوتر المضمنة والمكتبية، والتي تركز بشكل أكبر على زمن الاستجابة، مقابل الخوادم، التي تركز بشكل أكبر على الإنتاجية Throughput.

• عند مناقشة أداء أجهزة الكمبيوتر، سنهتم بشكل أساسي بزمن الاستجابة. و بالتالي لتحقيق أقصى قدر من الأداء،

نريد تقليل **زمن الاستجابة Response Time** أو **زمن التنفيذ Execution Time** لبعض المهام وبالتالي، يمكننا ربط

الأداء Performance و زمن التنفيذ لجهاز كمبيوتر X بالعلاقة التالية: $Performance_x = \frac{1}{Execution\ time_x}$

• هذا يعني أنه بالنسبة إلى جهازي كمبيوتر X و Y، إذا كان أداء X أكبر من أداء Y، فلدينا:

$$Performance_x > Performance_y$$
$$\frac{1}{Execution\ time_x} > \frac{1}{Execution\ time_y}$$
$$Execution\ time_y > Execution\ time_x$$

• أي أن زمن التنفيذ على Y أطول منه في X، إذا كان X أسرع من Y

• عند مناقشة تصميم الكمبيوتر، غالبًا ما نريد مقارنة أداء جهازي كمبيوتر مختلفين من الناحية الكمية.

سنستخدم العبارة X أسرع بمقدار n مرة من Y أو بالمقابل X أسرع بمقدار n مرة من Y أي:

$$\frac{\text{Performance}_X}{\text{Performance}_Y} = n$$

• إذا كانت X أسرع من Y بمقدار n مرة، فإن زمن التنفيذ على Y يكون أطول منه في X بمقدار n مرة

$$\frac{\text{Performance}_X}{\text{Performance}_Y} = \frac{\text{Execution time}_Y}{\text{Execution time}_X} = n$$

• إذا قام الكمبيوتر "A" بتنفيذ برنامج في 10 ثوانٍ وكان الكمبيوتر "B" يقوم بتنفيذ نفس البرنامج في 15 ثانية ، فما مقدار

سرعة "A" عن "B" ؟

• الحل:

$$\frac{\text{Performance}_A}{\text{Performance}_B} = \frac{\text{Execution time}_B}{\text{Execution time}_A} = n \quad \text{نعلم أن A أسرع بمقدار } n \text{ مرة من B إذا:}$$

$$\frac{15}{10} = 1.5 \quad \text{وبالتالي فإن نسبة الأداء:}$$

وبالتالي فإن A أسرع بمقدار 1.5 مرة من B .

$$\frac{\text{Performance}_A}{\text{Performance}_B} = 1.5 \quad \text{يمكننا القول أيضًا أن الكمبيوتر B أبطأ بمقدار 1.5 مرة من الكمبيوتر A، بما أن:}$$

$$\frac{\text{Performance}_A}{1.5} = \text{Performance}_B \quad \text{مما يعني:}$$

قياس الأداء

- الزمن هو مقياس أداء الكمبيوتر.
- الكمبيوتر الذي ينجز نفس العمل في أقل زمن هو الأسرع.
- يتم قياس زمن تنفيذ البرنامج بوحدة **الثواني للبرنامج** (seconds per program).
- يشمل الزمن الكلي لإكمال المهمة عمليات الوصول إلى القرص، والوصول إلى الذاكرة، وعمليات الإدخال/الإخراج، وعمليات نظام التشغيل.
- غالبًا ما تتم مشاركة أجهزة الكمبيوتر، وقد يعمل المعالج على عدة برامج في وقت واحد. في مثل هذه الحالات، قد يحاول النظام تحسين الوضع بدلاً من محاولة تقليل الزمن المنقضي لبرنامج واحد. لذلك يجب التمييز بين زمن تنفيذ وحدة المعالجة المركزية و الزمن الكلي لإكمال أي مهمة.
- **زمن تنفيذ وحدة المعالجة المركزية (CPU time)** : هو الزمن الذي تستغرقه وحدة المعالجة المركزية في الحوسبة لهذه المهمة ولا يشمل الزمن الذي يقضيه في انتظار الإدخال/الإخراج أو تشغيل برامج أخرى.

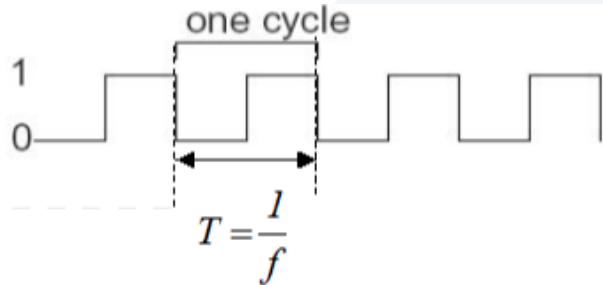
أداء وحدة المعالجة المركزية وعواملها

- غالبًا ما يقوم المستخدمون والمصممون بفحص الأداء باستخدام مقاييس مختلفة.
- إذا تمكنا من ربط هذه المقاييس المختلفة، فيمكننا تحديد تأثير تغيير التصميم على الأداء كما يختبره المستخدم.
- نظرًا لأننا نقصر أنفسنا على أداء وحدة المعالجة المركزية في هذه المرحلة، فإن مقياس الأداء الأساسي هو زمن تنفيذ وحدة المعالجة المركزية (CPU execution time).
- هناك علاقة بسيطة تربط المقاييس الأساسية (دورات الساعة clock cycles وزمن دورة الساعة clock cycle time) بزمن وحدة المعالجة المركزية CPU time.

$$\text{CPU execution time for a program} = \text{CPU clock cycles for a program} \times \text{Clock cycle time}$$

$$\text{CPU execution time for a program} = \frac{\text{CPU clock cycles for a program}}{\text{Clock rate}}$$

حيث:



• clock cycle : هو زمن دور ساعة واحد T

• clock rate : هو مقلوب الدور و يعطى بالعلاقة $f = \frac{1}{T}$

- توضح الصيغة الرياضية السابقة أن مصمم الأجهزة يمكنه تحسين الأداء عن طريق تقليل عدد دورات الساعة المطلوبة لبرنامج (CPU clock cycles for a program) أو تقليل زمن دورة الساعة (Clock cycle time).
- غالبًا ما يواجه المصمم مفاضلة بين عدد دورات الساعة اللازمة للبرنامج و زمن دورة الساعة. قد تؤدي أيضًا العديد من التقنيات التي تقلل عدد دورات الساعة إلى زيادة زمن دورة الساعة.



$$\text{CPU time}_A = \frac{\text{CPU clock cycles}_A}{\text{Clock rate}_A}$$

$$10 \text{ seconds} = \frac{\text{CPU clock cycles}_A}{2 \times 10^9 \frac{\text{cycles}}{\text{second}}}$$

$$\text{CPU clock cycles}_A = 10 \text{ seconds} \times 2 \times 10^9 \frac{\text{cycles}}{\text{second}} = 20 \times 10^9 \text{ cycles}$$

CPU time for B can be found using this equation:

$$\text{CPU time}_B = \frac{1.2 \times \text{CPU clock cycles}_A}{\text{Clock rate}_B}$$

$$6 \text{ seconds} = \frac{1.2 \times 20 \times 10^9 \text{ cycles}}{\text{Clock rate}_B}$$

$$\text{Clock rate}_B = \frac{1.2 \times 20 \times 10^9 \text{ cycles}}{6 \text{ seconds}} = \frac{0.2 \times 20 \times 10^9 \text{ cycles}}{\text{second}} = \frac{4 \times 10^9 \text{ cycles}}{\text{second}} = 4 \text{ GHz}$$

To run the program in 6 seconds, B must have twice the clock rate of A.

- بفرض لدينا برنامج معين يعمل في 10 ثوانٍ على الكمبيوتر A، الذي يعمل على نبضات ساعة 2 جيجا هرتز. نحاول مساعدة مصمم الكمبيوتر على بناء جهاز كمبيوتر B، والذي سيقوم بتشغيل هذا البرنامج في 6 ثوانٍ. قرر المصمم أن زيادة كبيرة في معدل (تردد) الساعة أمر ممكن، ولكن هذه الزيادة ستؤثر على بقية تصميم وحدة المعالجة المركزية، مما يتسبب في أن يتطلب الكمبيوتر B 1.2 ضعف عدد دورات الساعة التي يتطلبها الكمبيوتر A لتنفيذ نفس البرنامج. ما هو معدل (تردد) الساعة الذي يجب أن نطلب من المصمم تحقيقه؟



جامعة
المنارة

معادلة الأداء الأساسية

- لم تتضمن معادلات الأداء الواردة سابقاً أي إشارة إلى **عدد التعليمات** اللازمة للبرنامج (Number of instructions).
- بما أنه يجب على الكمبيوتر تنفيذ تعليمات البرنامج، لذلك يعتمد زمن التنفيذ على **عدد التعليمات** في البرنامج.
- يمكن النظر لزمن التنفيذ على أنه يساوي عدد التعليمات المنفذة مضروباً في متوسط الزمن للتعليمات الواحدة.
- لذلك، يمكن كتابة عدد دورات الساعة المطلوبة لبرنامج ما كما يلي:

$$\text{CPU clock cycles} = \text{Instructions for a program} \times \text{Average clock cycles per instruction}$$

حيث :

clock cycles per instruction : متوسط عدد دورات الساعة التي تحتاجها كل تعليمة لتنفيذها و غالباً ما يتم اختصاره بالمصطلح CPI

- يمكننا الآن كتابة معادلة الأداء الأساسية بدلالة عدد التعليمات التي ينفذها البرنامج (instruction count) ، و متوسط عدد دورات الساعة التي تحتاجها كل تعليمة لتنفيذها (CPI) و زمن دورة الساعة (clock cycle time):

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

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



Example

- يحاول مصمم برامج أن يختار بين تسلسلين للتعليمات البرمجية لجهاز كمبيوتر معين. حيث تتوفر ثلاث أصناف من التعليمات البرمجية (A و B و C) موضحة في الجدول التالي مع عدد أدوار الساعة التي تحتاجها تعليمة من كل صنف لكي تنفذ :

	CPI لكل صنف من التعليمات		
	A	B	C
CPI	1	2	3

- قام المصمم باقتراح تسلسلين من التعليمات البرمجية (التسلسل 1 و التسلسل 2) حيث يوضح الجدول التالي عدد التعليمات البرمجية المستخدمة من كل صنف من أصناف التعليمات في كلا التسلسلين:

تسلسل التعليمات	عدد التعليمات المستخدمة من كل صنف		
	A	B	C
1	2	1	2
2	4	1	1

- ما هو تسلسل البرنامج الذي ينفذ تعليمات أكثر؟ أيهما سيكون أسرع؟
- ما هو CPI (عدد أدوار الساعة التي تحتاجها كل تعليمة في التسلسلين)؟



جامعة
المنارة
MANARA UNIVERSITY

الحل

- التسلسل 1 ينفذ $2 + 1 + 2 = 5$ تعليمات.
- التسلسل 2 ينفذ $6 = 1 + 1 + 4$ تعليمات. و بالتالي، ينفذ التسلسل 1 تعليمات أقل.
- لإيجاد العدد الإجمالي لدورات الساعة لكل تسلسل نستخدم العلاقة التالية:

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

$$\text{CPU clock cycles}_1 = (2 \times 1) + (1 \times 2) + (2 \times 3) = 2 + 2 + 6 = 10 \text{ cycles}$$

$$\text{CPU clock cycles}_2 = (4 \times 1) + (1 \times 2) + (1 \times 3) = 4 + 2 + 3 = 9 \text{ cycles}$$

لذا فإن تسلسل الكود 2 أسرع، على الرغم من أنه ينفذ تعليمة إضافية واحدة.

نظرًا لأن تسلسل الكود 2 يتطلب عددًا أقل من دورات الساعة الإجمالية ولكنه يحتوي على تعليمات أكثر، فيجب أن يكون له CPI أقل. يمكن حساب قيم CPI كما يلي:

$$\text{CPI} = \frac{\text{CPU clock cycles}}{\text{Instruction count}}$$

$$\text{CPI}_1 = \frac{\text{CPU clock cycles}_1}{\text{Instruction count}_1} = \frac{10}{5} = 2.0$$

$$\text{CPI}_2 = \frac{\text{CPU clock cycles}_2}{\text{Instruction count}_2} = \frac{9}{6} = 1.5$$