❤ Java ❤

المحاضرة السابعة   الأحد   22/10/2017

تقدير كلفة الخوارزمية: غالباً ما نقدر الكلفة تقدير وغالباً ما يكون تقديرنا صحيح وخاصة أننا ننظر للكلفة من O(f)

لنكن لدينا الخوارزمية التالية:

```
for(int i=0; i<n-1; i++)
    for(int j=i+1; j<n; j++)
        if (A[i] > A[j])
        {swap(A[i], A[j]);
        }
```

swap دالة التبديل
تقوم هذه الخوارزمية بترتيب عناصر المصفوفة ترتيب تصاعدي

إن تقدير كلفة الخوارزمية السابقة بالنسبة لعدد عمليات ال swap هو:

$$\underbrace{n}_{\text{تقدير عدد مرات دخول الحلقة for الأولى}} \times \underbrace{n}_{\text{تقدير عدد مرات دخول حلقة for الثانية}} \times \underbrace{1}_{\text{عدد عمليات ال swap}} = n^2 \qquad O(n^2)$$

* كلفتها الحقيقية بالنسبة لعدد عمليات ال swap هي:

i تأخذ القيم: 0   1   ....   n-2

j: n-1   n-2   1

$$\sum_{i=0}^{n-2} i+1 = \sum_{i=0}^{n-2} j = \frac{(n-2)(n-1)}{2} = \frac{n^2-3n+2}{2}$$

$$= \frac{1}{2}n^2 - \frac{3}{2}n + 1 \qquad O(n^2)$$

نلاحظ أنه بالتقدير والحقيقية فإن $O(n^2)$ أي أننا لازلنا في نفس الصف أي أن الخوارزمية كلفتها تربيعية

كنا ننظر لـ n كبيرة جداً جداً أي n ≈ n-3, n-5 عند الـ ∞ نفس الشيء
لذلك لم نهتم بالاختلاف بين التقدير والفعلي عملياً دائماً في الـ O(f)

كلفة الحلقات المتداخلة: عدد تكرارات الحلقة الأولى × عدد تكرارات الحلقة الثانية × .... × عدد تكرارات الحلقة الأخيرة × عدد العمليات المطلوبة الكلفة بالنسبة لها الموجودة في الحلقة الأخيرة.

الخوارزميات العودية:

تعريف عودي لمجموعة الأعداد الطبيعية: حسب العالم بيانو:

* الصفر عدد طبيعي
* كل عدد يلي عدد طبيعي هو عدد طبيعي (يلي يعني +1)

$$\mathbb{N} = \{0, 1, 2, 3, \dots\}$$

* نقول عن شيء ما أو هدف ما أنه عودي إذا عُرِّف كلياً أو جزئياً بدلالة نفسه.
* العودية ليست فقط تعريف رياضي إنما هي تمثيل لظواهر طبيعية مثل الصدى وانعكاس الصورة على مرآتين متقابلتين
* الشيء يكرر نفسه وفي كل تكرار يصغر حتى يصل لحالة التوقف

حالة التوقف: الحالة التي لا تكرر نفسها وهي الحالة التي نعرف عندها النتيجة فوراً.

مثال: هل $3 \in \mathbb{N}$ ؟؟

نحن نعلم أن $0 \in \mathbb{N}$ (حالة توقف) نعرف الجواب

prev(3) = 2 $\overset{?}{=}$ 0 ← 3 عدد طبيعي

prev(2) = 1 $\overset{?}{=}$ 0 ← 2 عدد طبيعي

prev(1) = 0 = 0 ← عدد طبيعي

1 عدد طبيعي

مثال: مضاريب n أو n! يعرف بالشكل: ((رياضي))

$$0! = 1$$

$$n! = n \times \underbrace{n-1 \times n-2 \times \dots \times 2 \times 1}_{(n-1)!}$$

التعريف العودي:

$$n! = \begin{cases} 1 & : n = 0 \\ n(n-1)! & : n > 0 \end{cases}$$

لماذا الخوارزميات العودية؟؟ * لأن كتابتها أسهل ((كتابة الخوارزمية العودية مطابق للتعريف الرياضي تماماً))

```
int fact(int n){ if (n==0) return (0);
                 else return (n*fact(n-1));
}
```

* كثير من المسائل تحمل في داخلها الظاهرة العودية فالأسهل حلها

بالطريقة العودية وليس بالطريقة التكرارية.

* الخوارزمية التكرارية هي الخوارزمية التي تعتمد على الحلقات التكرارية.
* الخوارزمية العودية هي خوارزمية تستدعي نفسها

مساوئ الخوارزميات العودية:

* تنفيذها طويل ويتم على طورين طور الاستدعاءات ومن ثم طور تجميع النتائج

Fact(4)

24 = 4 * Fact(3)

6 = 3 * Fact(2)

2 = 2 * Fact(1)

1 = 1 * Fact(0)

استدعاءات

تجميع النتائج

= 1 حالة توقف

أي انتهى طور الاستدعاءات

عدد مرات استدعاء الخوارزمية لنفسها يسمى العمق العودي
(Recurrisf Depth)

وهو مهم جداً في الخوارزميات العودية

* لا استطيع حساب النتيجة إلا عندما أصل إلى حالة التوقف ولذلك فإن الاستدعاءات جميعها تبقى محجوزة في الذاكرة إلى أن ينتهي التنفيذ
* تستهلك الخوارزميات العودية ذاكرة كبيرة قد تكون غير متوفرة لذلك يجب أن نوازن بين سهولة كتابة البرنامج وحل المسألة وبين كمية الذاكرة المستهلكة.

حساب كلفة الخوارزمية العودية:

سنرى كيف نحسب كلفة الخوارزمية العودية. هل هناك جدوى منها؟ هل هي نفس الخوارزمية التكرارية أم أقل منها أم أكثر؟؟

* إذا عرفنا العمق العودي يصبح حساب الكلفة سهل جداً

عدد العمليات المطلوب حساب كلفتها (في كل استدعاء) × العمق العودي ⇐ نحصل على الكلفة المطلوبة

حساب الكلفة بالنسبة لعمليات الضرب في fact(n), n!

العمق العودي = n ، في كل استدعاء عملية ضرب واحدة:

⇐ عدد عمليات الضرب = n أي $T(n) = n$

في هذا المثال كان من السهل معرفة العمق العودي ولكن حساب العمق العودي ليس بهذه السهولة.

* في حال عدم معرفة العمق العودي نحسبها بطريقة تراجعية

$T(n) = 1 + T(n-1)$ ((معادلة تراجعية))

عدد العمليات في الاستدعاء الواحد

$= 1 + [1 + T(n-2)]$

$= \dots = \underbrace{1+1+\dots+1}_{n-1 \text{ مرة}} + T(1) = \underbrace{1+1+\dots+1}_{n \text{ مرة}} + T(0)$

$= n + \underline{T(0)} = n$

0 ((لا يوجد أي عملية ضرب))

كم كلفة الذاكرة؟؟ أي كم متحول نحتاج أن نحجز كي تتنفذ الخوارزمية

الكلفة بالنسبة للذاكرة

$S(n) = \underline{n} \times \text{sizeof(int)}$

نحتاج n استدعاء في كل استدعاء متحول من النوع الصحيح

الحل بالطريقة التكرارية:

```
int f(int n){
        int R=1;
    for(int i=1; i<=n; i++)
            R=R*i;
    return R;
}
```

$S(n) = 2 \times \text{sizeof(int)}$, $T(n) = n$ عملية ضرب

* لا فرق في هذه المسألة بين العودي والتكراري في تكلفة الخوارزمية بالعكس هناك كلفة ذاكرة إضافية في العودي و الحقيقة أن أهمية العودية لا تظهر في مثل هذه المسائل.

* هذا المثال كان حجة بعض المبرمجين للحكم على الخوارزميات العودية بعدم الفائدة

وظيفة: اكتب خوارزميات عودية تقوم بمايلي:-

1- جمع عناصر متجهة 2- جداء عناصر متجهة 3- ضرب عددين صحيحين

4- قسمة عددين صحيحين 5- البحث عن دليل أصغر عنصر في متجهة

6- البحث عن دليل أكبر عنصر في متجهة

انتهت المحاضرة