

Soft Problem

نحوه مربکاری

Sorting Problem

- دنبالهای سالم و نامنظم

$$A = \langle a_1, a_2, \dots, a_n \rangle$$

خاصیت دنباله را به معنی بازآرایی نمود

$$A' = \langle a'_1, a'_2, \dots, a'_n \rangle$$

$$a'_i \leq a'_{i+1}$$

بازآرایی هر کدامی باشد $i = 1, 2, \dots, n-1$

- مکرر سازی را فیکان در فرم $\overbrace{m \times m}$ باشد:

هارای (n) دوست $A = \langle a_1, \dots, a_n \rangle$ باشد

جای σ را فصل جای σ باشد

لعنی بر روی مجموعه S^n باشد $\{\sigma_1, \sigma_2, \dots, \sigma_n\}$

بازای $a_{\sigma_i} < a_{\sigma_{i+1}}$ باشد

$$A = \langle 4, 3, 7, 2, 1, 5 \rangle \cdot \underline{\text{جاء}}^9$$

$$A' = \langle \hat{1}, \hat{2}, \hat{3}, \hat{4}, \hat{5}, \hat{7} \rangle$$

$$\sigma = \langle 0, 5, 2, 1, 4, 3 \rangle$$

$$\min \sum |a_{\sigma_{i+1}} - a_{\sigma_i}|$$

النهاية

متحدة

مع وجوه رائمه

الگوریتم های مرتب سازی

Bubble Sort - مرتب سازی حبابی

Select Sort - مرتب سازی انتخابی $O(n^2)$

Insert Sort - مرتب سازی درجی

Merge Sort - مرتب سازی ارتدامی

Quick Sort - مرتب سازی سریع $\star O(n \log n)$

Heapsort - مرتب سازی هیپ

Pseudo Code نمبر ۸

- اعلان نوع متغيرها صناعي و غير صناعي
- اعيار رسمية نوع بياناتها دير ورودي صريطي
- توليد خروج زباونهم
- فرض حجم وابع خط بلا درسته هست
- هدف ، ارائه اوندو منطق طلي حکم بر الگوریتم است.

BubbleSort(A)

n = length(A)

for r = 1 to n-1

 for i = 1 to n-1

 if A[i+1] < A[i]

 swap(A[i], A[i+1])

BubbleSort(A)

n = length(A)

for r = 1 to n-1

 for i = 1 to n-r

 if a[i+1] < a[i]

 swap(a[i], a[i+1])

BubbleSort(A)

n = length(A)

for r = 1 to n-1

 sorted = True

 for i = 1 to n-r

 if a[i+1] < a[i]

 swap(a[i], a[i+1])

 sorted = False

 if sorted then break

آنالیز مرتب سازی مبتنی

BubbleSort(A)

n = length(A)

for r = 1 to n-1

 for i = 1 to n-1

 if a[i+1] < a[i]

 swap(a[i], a[i+1])

 هر چند بار اجرا
 تعداد بارهای اجرا

cost	time
c_1	1
c_2	$n-1$
c_2	$(n-1)^2$
c_3	$(n-1)^2$
c_4	t_1

for $r=1$ to $n-1$



$c_1 \quad c_2 \quad c_3$

for($r=1$; $r < n$; $r++$)

$\sim \sim \sim \sim$

$c_1 \quad , \underline{w}$

$c_2 \quad , \underline{n}$

$c_3 \quad , \underline{n-1}$

آنالیز مرتب سازی جعبه - آئینه حالت - Best Case

آنالیز با بررسی، نموده و در آن اگر آرایه میان قسم را صرف نماید.

BubbleSort(A)

$n = \text{length}(A)$

for $r=1$ to $n-1$

 for $i=1$ to $n-1$

 if $a[i+1] < a[i]$

 swap($a[i], a[i+1]$)

cost	time
c_1	1
c_2	$n-1$
c_2	$(n-1)^2$
c_3	$(n-1)^2$
c_4	$t_1 = 0$

- در این الگوریتم ورودی که به آرایه آن، سه کوچکتر است: لینی ورودی از آب سرمه و دو دست.

آنالیز مرتب سازی مجاہد - Best Case این حالت با برخوبی، نونه ۱۰۰٪ این الگوریتم را در آن کمین زمان قابل راه رفتن

BubbleSort(A)

$n = \text{length}(A)$

for $r=1$ to $n-1$

 for $i=1$ to $n-1$

 if $a[i+1] < a[i]$

 swap($a[i], a[i+1]$)

cost	time
c_1	1
c_2	$n-1$
c_2	$(n-1)^2$
c_3	$(n-1)^2$
c_4	$t_i = 0$

$$T(n) = c_1 + c_2(n-1) + (c_2 + c_3)(n-1)^2 = O(n^2)$$

آنالیز مرتب سازی مجاہد - بدین حالت - Worst Case

بدین طبقه با برخوبی، نونه ۱۱۰۰ کی ایجادی سی الگوریتم است که به زیرا این

بسیار زمان طلب را صرف می کند.

BubbleSort(A)

$n = \text{length}(A)$

for $r = 1$ to $n-1$

 for $i = 1$ to $n-1$

 if $a[i+1] < a[i]$

 swap($a[i], a[i+1]$)

cost	time
c_1	1
c_2	$n-1$
c_2	$(n-1)^2$
c_3	$(n-1)^2$
c_4	$t_1 = (n-1)^2$ *

$$T(n) = c_1 + c_2(n-1) + (c_2 + c_3 + c_4)(n-1)^2 = O(n^2)$$

8	7	6	5	4	3	2	1
---	---	---	---	---	---	---	---

7	6	5	4	3	2	1	8
---	---	---	---	---	---	---	---

6	5	4	3	2	1	7	8
---	---	---	---	---	---	---	---

$n-1$

$+ n-2$

$+ n-3$

\dots

$+$

\vdots

$+ 1$

$\mathcal{O}(n^2)$

$$\Rightarrow t_1 = \frac{n(n-1)}{2}$$

$$n(n-1)/2$$

$$\Rightarrow T(n) = O(n^2)$$