طراحی و تحلیل الگوریتم

استاد:

دکتر زاهد رحمتی

تدریسیاران:

داریوش کاظمی

اشکان ودادی

ترم دوم ۱۴۰۰

Assignments

30% of Score

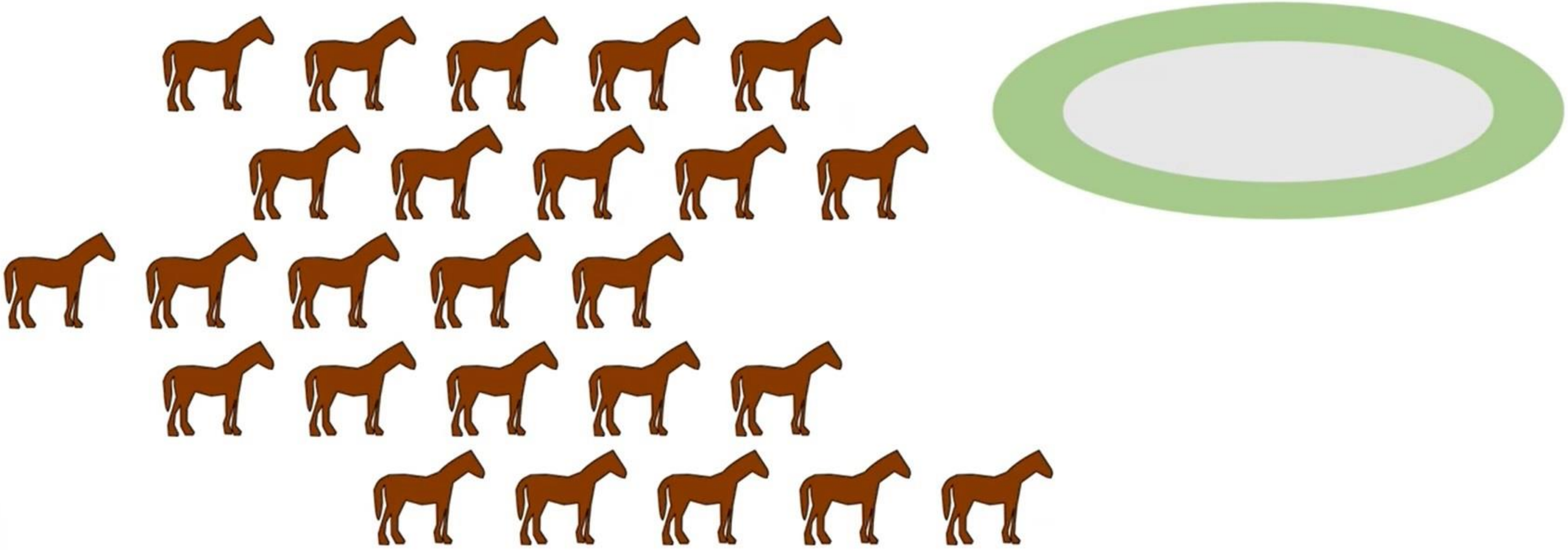
- 6 assignments, each worth 5% of score

1. Algorithm and time Complexity
2. …
3. …
4. …
5. …
6. …

Interview Question

There are 25 horses. What is the minimum number of races needed so you can identify the fastest 3 horses? You can race up to 5 horses at a time, but you do not have a watch.

What is the minimum number of races you need to identify the fastest 3 of all the horses?



How do you solve it?

You can find the 3 fastest horses in a minimum of **7 races**.

<u>Step one</u>: Divide the 25 horses into groups of 5, and race the horses in each group. **(5 races)**

<u>Step two</u>: Take the winner from each group and race those 5 horses. The winner of this race (the winner of the winners) is the fastest horse overall. (**1 race**)

> Label the 5 groups from step one as a, b, c, d, e to correspond to the groups for the horses that finished 1st, 2nd, 3rd, 4th, and 5th for the race in step two.
>
> Write a subscript to identify the order that the horse finished in the group, so a_2 means the horse that finished 2nd place in group a.

<u>Step three</u>: Do one more race with a_2, a_3, b_1, b_2, c_1. That is, race the second and third fastest horses from a, the fastest and second fastest from b, and the fastest from c. The fastest 2 horses are the 2nd and 3rd fastest horses overall. (**1 race**)

Some inspiring stories

George B. Dantzig

Timothy M. Chan

جلسه اول

روش جایگزینی در حل مسائل بازگشتی

(بخش ۴-۳ کتاب CLRS)

حل مسائل بازگشتی

1. Substitution method
2. Iterating the recurrence (Brute Force)
3. Recursion Tree
4. Master Theorem

حل مسائل بازگشتی

1. **Substitution method**
2. Iterating the recurrence (Brute Force)
3. Recursion Tree
4. Master Theorem

Substitution method:

1. Guess
2. Verify
3. Solve

We make a guess for the solution and then we use mathematical induction to prove the guess is correct or incorrect.

حل مسائل بازگشتی

Substitution method:

- In the substitution method, instead of trying to find an exact closed-form solution, we only try to find a closed-form bound on the recurrence. This is often much **easier than finding a full closed-form solution**, as there is much greater leeway in dealing with constants.

- The substitution method is a powerful approach that is able to prove upper bounds **for almost all recurrences**. However, its power is not always needed.

- Note that the substitution method still requires the use of **induction**.

حل مسائل بازگشتی

Substitution method:

چه طور یک حدس خوب بزنیم؟

- یکی از چالش‌های این روش حدس خوب و دقیق است. روش نوشته شده ای برای حدس زدن نیست ولی درکل برای حدس خوب می‌توان:

1. تمرین زیاد حل کنیم (کتاب CLRS)
2. با کمک روش‌های تکرار با جایگزینی (iterative) حدود دستمان بیاید
3. با کمک رسم درخت بازگشت حدود دستمان بیاید
4. هر حدسی که زدیم مراحل استقرا را دقیق انجام دهیم تا اگر جایی قرار است رد شود بتوانیم رد کنیم

سوال1: جواب $T(n)=T(n-1)+n$ را نشان بدهید؟ $O(?)$

سوال1: جواب T(n)=T(n-1)+n را نشان بدهید؟ $O(n^2)$

سوال1: جواب T(n)=T(n-1)+n را نشان بدهید؟ O(n^2)

Guess: $T(n) \le cn^2$, $c = \max\big(1, T(1)\big)$

$$
\begin{aligned}
T(n) &= T(n-1) + n \quad \textit{(problem)} \\
&\le c(n-1)^2 + n \quad \textit{(inductive hypothesis)} \\
&= cn^2 + (1-2c)n + 1 \ (n \ge 1 \ \rightarrow \min(n) = 1) \\
&\le cn^2 + 2 - 2c \quad (2 - 2c \le 0) \\
&\le \mathrm{cn}^2 \quad (c \ge 1)
\end{aligned}
$$

برای شهود کافی است، در هر مرحله T(n−k) را باز کنید که به صورت مجموعه n+…+1 درمی‌آید که همان جواب است.

سوال 2: جواب $T(n) = 2T(\left\lfloor \frac{n}{2} \right\rfloor) + n$ را نشان بدهید؟ O(?)

سوال 2: جواب $T(n) = 2T(\left\lfloor \frac{n}{2} \right\rfloor) + n$ را نشان بدهید؟ O(?)

دام: جواب غلط را میشود O(n) بدست آورد.

Guess: $T(n) \le cn$

$T(n) = 2T(\left\lfloor \frac{n}{2} \right\rfloor) + n \quad (problem)$

$\le 2\left(c\left\lfloor \frac{n}{2} \right\rfloor\right) + n$

$\le cn + n$

$= O(n) \qquad wrong\ answer$

چرا؟

سوال 2: جواب n+ $T(n) = 2T(\left\lfloor\frac{n}{2}\right\rfloor)$ را نشان بدهید؟ O(?)

دام: جواب غلط را میشود O(n) بدست آورد.

Guess: $T(n) \leq cn$

$$T(n) = 2T(\left\lfloor\frac{n}{2}\right\rfloor) + n \quad \textit{(problem)}$$
$$\leq 2\left(c\left\lfloor\frac{n}{2}\right\rfloor\right) + n$$
$$\leq cn + \mathrm{n}$$
$$= O(n) \qquad \textit{wrong answer}$$

چرا؟

شکل دقیق فرض استقرا را ثابت نکردیم. محدوده‌ی c باید معلوم باشد.

سوال 2: جواب $T(n) = 2T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + n$ را نشان بدهید؟ O(?)

سوال 2: جواب $T(n) = 2T(\lfloor \frac{n}{2} \rfloor) + n$ را نشان بدهید؟ O(nlogn)

Guess: $T(n) \le cnlogn\ , c \ge 0$

$T(n) = 2T(\lfloor \frac{n}{2} \rfloor) + n$ *(problem)*

$\le 2(c \lfloor \frac{n}{2} \rfloor \log(\lfloor \frac{n}{2} \rfloor) + n$ *(inductive hypothesis)*

$\le cnlog\left(\frac{n}{2}\right) + n$

$= cnlogn - cnlog2 + n$ (*mathematic operation*)

$= cnlogn\ - cn + n\ (c \ge 1)$

$\le cn \log n$

کامل اثبات استقرای مرزی را در clrs مطالعه کنید. (بخش 4-3)

سوال 3: جواب $T(n) = T(\left\lceil \frac{n}{2} \right\rceil) + 1$ را نشان بدهید؟ O(?)

سوال 3: جواب $T(n) = T(\left\lceil \frac{n}{2} \right\rceil) + 1$ را نشان بدهید؟ O(logn)

سوال 3: جواب $T(n) = T(\left\lceil \frac{n}{2} \right\rceil) + 1$ را نشان بدهید؟ O(logn)

Guess: $T(n) \le clogn - 1\ , c = 3$

$T(n) = T(\left\lceil \frac{n}{2} \right\rceil) + 1$ *(problem)*

$\le 3\log\left(\left\lceil \frac{n}{2} \right\rceil\right) - 1 + 1$ *(inductive hypothesis)*

$\le 3\log\left(\frac{3n}{4}\right)$ (*mathematic operation and speed of log*)

$= 3\log(n) + 3\log(\frac{3}{4})$ (*mathematic operation*)

$\le 3\log(n) + \log\left(\frac{1}{2}\right)$ (*mathematic operation*)

$= 3\log n - 1$

سوال 4: جواب $T(n) = 2T(n-1) + c_1, T(1) = 1$ را نشان بدهید؟

By expanding this out a bit (using the "iteration method"), we can guess that this will be $O(2^n)$.
To use the substitution method to prove this bound, we now need to guess a closed-form upper bound based on this asymptotic bound.
We will guess an upper bound of $k2^n - b$, where b is some constant.
We include the b in anticipation of having to deal with the constant c1 that appears in the recurrence relation, and because it does no harm.
In the process of proving this bound by induction, we will generate a set of constraints on k and b, and if b turns out to be unnecessary, we will be able to set it to whatever we want at the end.

Our property, then, is $T(n) \leq k2^n - b$, for some two constants k and b.
Base case: $n = 1. T(1) = 1 \leq k2^1 - b = 2k - b$. This is true as long as $k \geq (b + 1)/2$.
Inductive case: We assume our property is true for n – 1. We now want to show that it is true for n.

T(n) = 2T(n–1) + c1
≤ 2(k2^n – 1 – b) + c1 (by IH)
= k2^n – 2b + c1
≤ k2^n – b

This is true as long as b ≥ c1.

سوال 4: جواب $T(n) = 2T(n-1) + c_1, T(1) = 1$ را نشان بدهید؟

Our property, then, is $T(n) \leq k2^n - b$, for some two constants k and b.
Base case: $n = 1. T(1) = 1 \leq k2^1 - b = 2k - b$. This is true as long as $k \geq (b + 1)/2$.
Inductive case: We assume our property is true for n – 1. We now want to show that it is true for n.

T(n)	=	2T(n−1) + c1	
	≤	2(k2^n – 1 – b) + c1	(by IH)
	=	k2^n – 2b + c1	
	≤	k2^n – b	

This is true as long as b ≥ c1.

So we end up with two constraints that need to be satisfied for this proof to work, and we can satisfy them simply by letting b = c1 and k = (b + 1)/2, which is always possible, as the definition of O allows us to choose any constant. Therefore, we have proved that our property is true, and so T(n) is O(2n).

خسته نباشید!

داریوش کاظمی – اشکان ودادی