

作业订正

- 每周三发回的作业，订正之后，次周一交
 - 没做的题
 - 打三角的错题
 - 其他错题（自愿）

- 作业讲解
 - CS第1.1节问题9、13
 - CS第1.2节问题15
 - CS第1.3节问题6、9、14
 - CS第1.5节问题8、10、15

CS第1.1节问题13

- ... on Day i you **receive** twice as many pennies as you did on Day $i-1$... How many will you **have** on Day n ?
 - $1+2^1+\dots+2^{n-1}=2^n-1$
 - 加法原理

CS第1.2节问题15

- In how many ways can we pair up all the members of the club?

- 检查简单的情况: $n=2$ 时, 有3种

- 一般的情况: $(2n)!/(n!2^n)$

- 另一种思路:
$$\frac{\binom{2n}{2}\binom{2n-2}{2}\cdots\binom{2}{2}}{n!}$$

- ... we also determine who serves first for each pairing...

- 检查简单的情况: $n=2$ 时, 有12种

- 一般的情况: $(2n)!/(n!)$

- 另一种思路:
$$\frac{\binom{2n}{2}\binom{2n-2}{2}\cdots\binom{2}{2}}{n!} \cdot 2^n$$

CS第1.3节问题6

- $\left(\frac{n!}{n_1!n_2!\cdots n_k!}\right)$ 是 $(x_1 + x_2 + \cdots + x_k)^n$ 中 $x_1^{n_1} x_2^{n_2} \cdots x_k^{n_k}$ 的系数。

- 如何直观解释？

- 方法1: $\binom{n}{n_1} \binom{n-n_1}{n_2} \cdots = \left(\frac{n!}{n_1!n_2!\cdots n_k!}\right)$

- 方法2: 你能直接解释吗？

CS第1.5节问题8b

- How can you relate the number of ways of placing k red checkers and $n-1$ black checkers in a row to the number of k -element multisets of an n -element set?



- $m(1)=1$
- $m(2)=2$
- $m(3)=0$
- $m(4)=3$
- $m(5)=2$
- $m(6)=0$

CS第1.5节问题8c

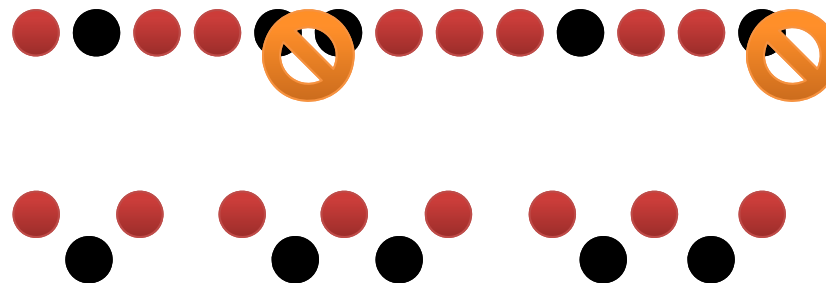
- How can you relate the choice of k items out of $n+k-1$ items to the placement of red and black checkers?



$$\frac{(n+k-1)!}{(n-1)!k!} = \binom{n+k-1}{k}$$

CS第1.5节问题10

- How many solutions to the equation $x_1+x_2+\dots+x_n=k$ are there with each x_i a **positive** integer?



CS第1.5节问题15

a. n^k

b. n^k

c. $\binom{n+k-1}{k}$

d. $\binom{n}{k}$

e. n^k

f. n^k

g. $\binom{n}{k}$

h. $\binom{n+k-1}{k}$

i. n^k

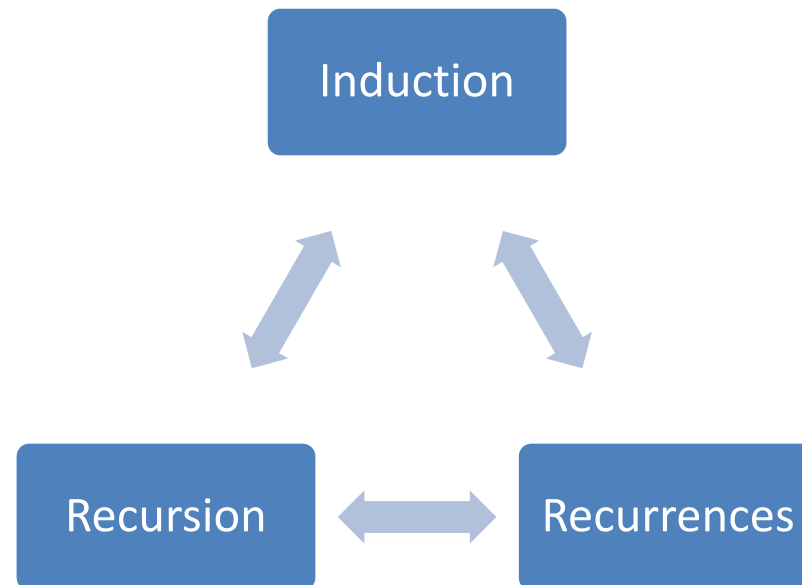
j. $\binom{n+k-1}{k}$

k. n^k

- 教材讨论
– CS第4章

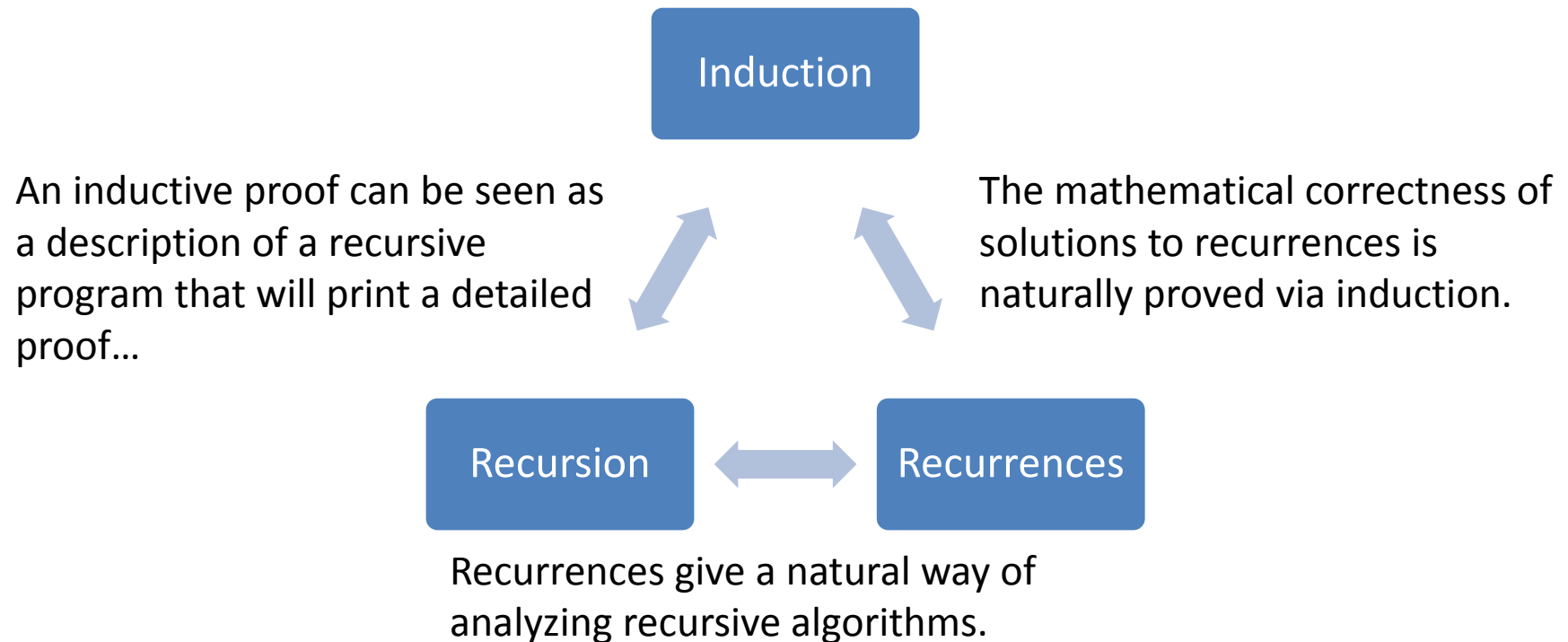
问题1: induction, recursion, recurrences

- 什么是induction、recursion、recurrences?
- 它们两两之间分别有什么联系?



问题1: induction, recursion, recurrences

- 什么是induction、recursion、recurrences?
- 它们两两之间分别有什么联系?



问题2: induction

- 什么是well-ordering principle?
- induction和well-ordering principle之间有什么联系?

问题2: induction (续)

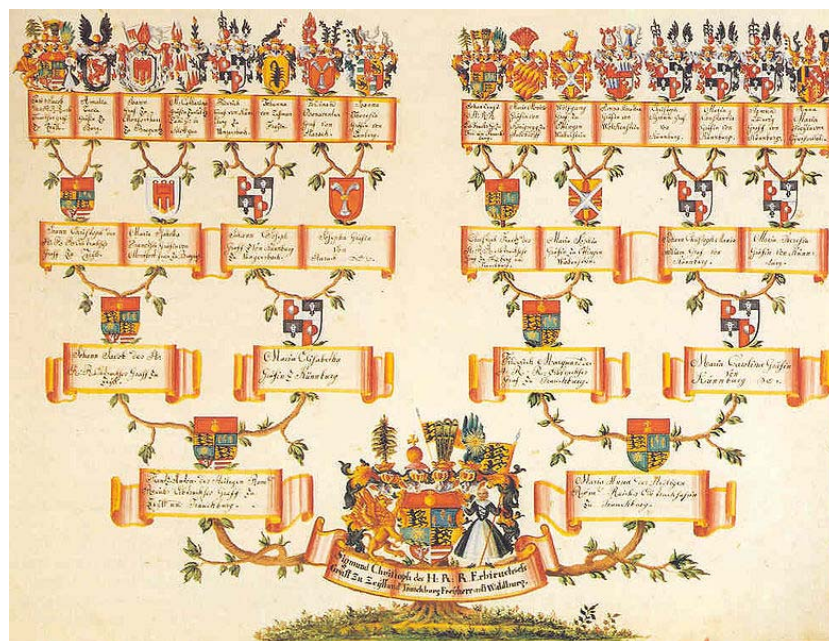
- 你怎么理解build a smaller case to a larger one (bottom-up) 和decompose the larger case into smaller ones (top-down)?
- 后者有哪些优势?

问题2: induction (续)

- 你怎么理解build a smaller case to a larger one (bottom-up) 和decompose the larger case into smaller ones (top-down)?
- 后者有哪些优势?
 - There are times when this way of thinking is clearly the best way to get a valid proof. Such examples occur throughout computer science.
 - “Building up” small cases into larger ones requires an additional step, namely showing that all larger cases can be created by using the construction given. It is often the case that a “building up” process constructs a proper subset of the possible cases.
 - With a top-down approach, there is no question about what our base case or base cases should be. The base cases are the ones where the recursive decomposition no longer works.

问题2: induction (续)

- 你怎么理解structural induction?
- 你能利用数学归纳法证明这个结论吗:
An ancestor tree extending over g generations shows at most $2^g - 1$ persons. (注意: 有些人的祖先信息可能缺失)



问题3: recurrences

- 当主定理不适用于一个递归式时，你还有什么手段？

$$T(n) = \begin{cases} rT(n-1) + g(n) & \text{if } n > 0 \\ a & \text{if } n = 0 \end{cases}$$

$$T(n) = 2T(n/4) + \sqrt{n} \lg n$$

问题3: recurrences

- 当主定理不适用于一个递归式时，你还有什么手段？

$$T(n) = \begin{cases} rT(n-1) + g(n) & \text{if } n > 0 \\ a & \text{if } n = 0 \end{cases}$$

$$T(n) = 2T(n/4) + \sqrt{n} \lg n$$

- 通过递推/递归树猜测，通过归纳法证明

问题3: recurrences (续)

- 你能为这个定理的证明绘制recursion tree吗?

Theorem 4.9 *Let a be an integer greater than or equal to 1 and b be a real number greater than 1. Let c be a positive real number and d a nonnegative real number. Given a recurrence of the form*

$$T(n) = \begin{cases} aT(n/b) + n^c & \text{if } n > 1 \\ d & \text{if } n = 1 \end{cases}$$

in which n is restricted to be a power of b ,

1. *if $\log_b a < c$, $T(n) = \Theta(n^c)$,*
2. *if $\log_b a = c$, $T(n) = \Theta(n^c \log n)$,*
3. *if $\log_b a > c$, $T(n) = \Theta(n^{\log_b a})$.*

问题4: recurrences and selection

- 第4.6节与前几节有什么联系?

```

Select1(A,i,n)
(selects the ith smallest element in set A, where n = |A| )
(1)  if (n = 1)
(2)      return the one item in A
(3)  else
(4)      p = MagicMiddle(A)
(5)      Let H be the set of elements greater than p
(6)      Let L be the set of elements less than or equal to p
(7)      if (i ≤ |L|)
(8)          Return Select1(L,i,|L|)
(9)      else
(10)         Return Select1(H,i - |L|,|H|).
    
```

$$T(n) \leq T(n/b) + cn.$$

- 你能解释MagicMiddle的基本思路吗?

```

MagicMiddle(A)
(1)  Let n = |A|
(2)  if (n < 60)
(3)      use sorting to return the median of A
(4)  else
(5)      Break A into k = n/5 groups of size 5, G1, ..., Gk
(6)      for i = 1 to k
(7)          find mi, the median of Gi (by sorting)
(8)      Let M = {m1, ..., mk}
(9)      return Select1 (M, [k/2], k).
    
```

