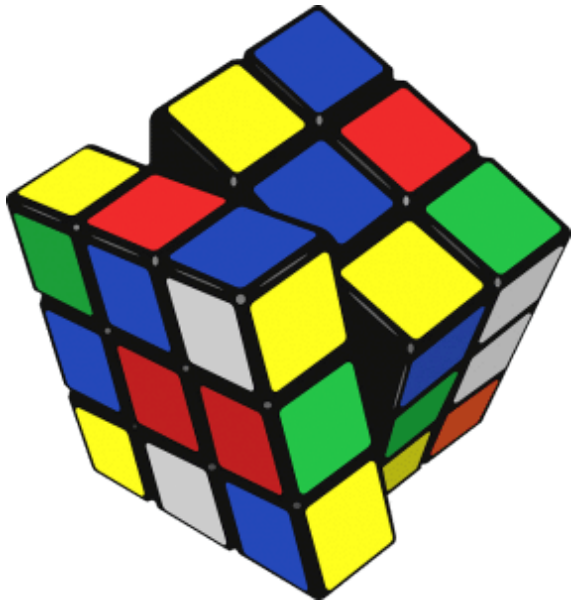
1-5 数据与数据结构

魏恒峰

hfwei@nju.edu.cn

2017 年 11 月 06 日

Permutations



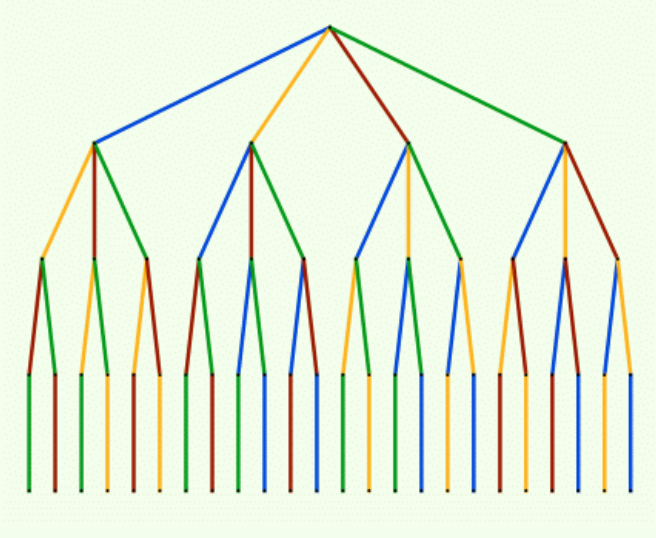
DH 2.9: # of Permutations

Prove that the number of permutations of $A_n = \{1 \cdots n\}$ is $n!$.

DH 2.9: # of Permutations

Prove that the number of permutations of $A_n = \{1 \cdots n\}$ is $n!$.

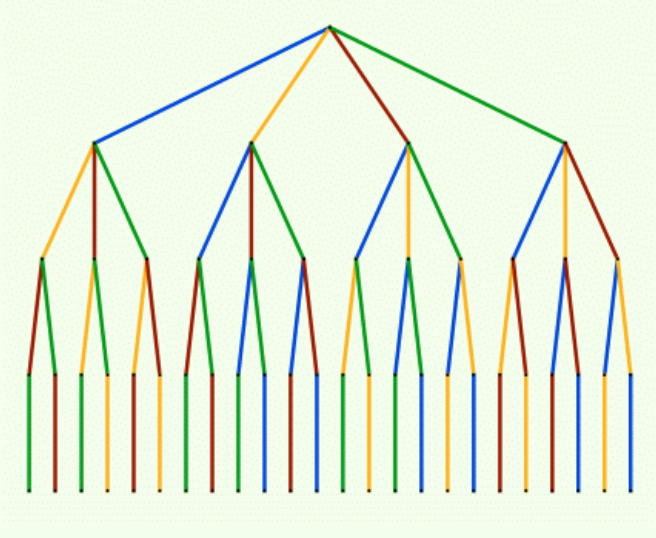
(1) The "choosing from" method:



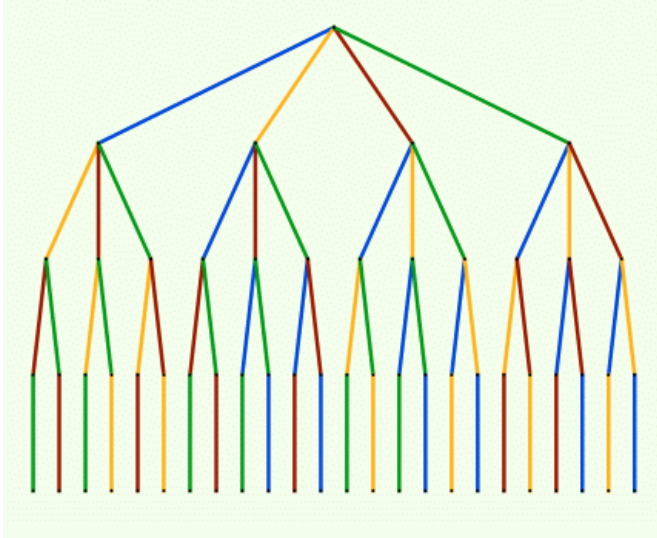
DH 2.9: # of Permutations

Prove that the number of permutations of $A_n = \{1 \cdots n\}$ is $n!$.

(1) The "choosing from" method:



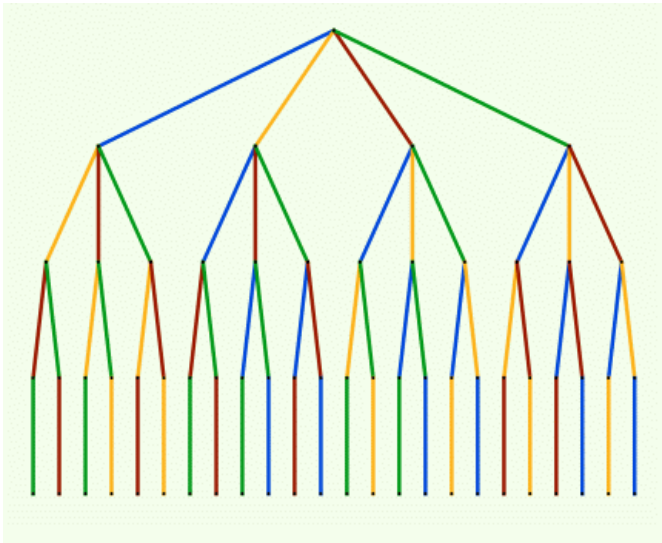
(2) The "inserting into" method:



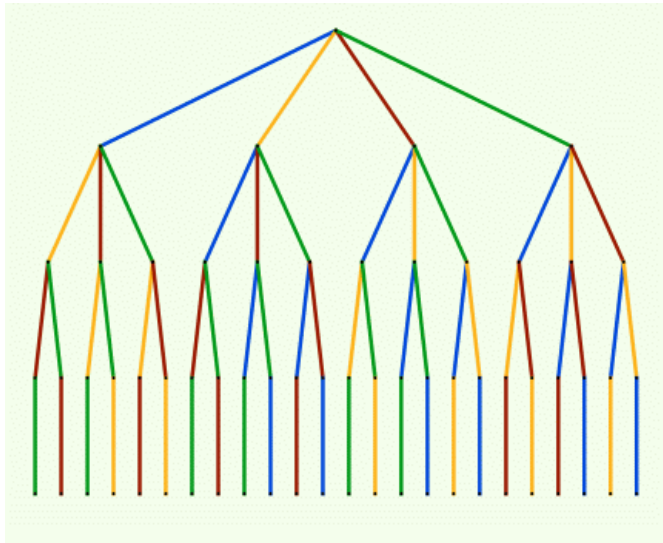
DH 2.9: # of Permutations

Prove that the number of permutations of $A_n = \{1 \cdots n\}$ is $n!$.

(1) The "choosing from" method:



(2) The "inserting into" method:



Prove by mathematical induction on n.

DH 2.9: # of Permutations

Prove that the number of permutations of $A_n = \{1 \cdots n\}$ is $n!$.

Prove by mathematical induction on n.

$$P(n) : \#\text{ of permutations of } A_n = \{1 \cdots n\} \text{ is } n!$$

DH 2.9: # of Permutations

Prove that the number of permutations of $A_n = \{1 \cdots n\}$ is $n!$.

Prove by mathematical induction on n.

$$P(n) : \#\text{ of permutations of } A_n = \{1 \cdots n\} \text{ is } n!$$

B.S. $P(1)$

I.H. $P(n)$

I.S. $P(n) \rightarrow P(n+1)$

DH 2.9: # of Permutations

Prove that the number of permutations of $A_n = \{1 \cdots n\}$ is $n!$.

Prove by mathematical induction on n.

$$P(n) : \#\text{ of permutations of } A_n = \{1 \cdots n\} \text{ is } n!$$

B.S. $P(1)$

I.H. $P(n)$

I.S. $P(n) \rightarrow P(n+1)$

DH 2.9: # of Permutations

Prove that the number of permutations of $A_n = \{1 \cdots n\}$ is $n!$.

Prove by mathematical induction on n.

$$P(n) : \#\text{ of permutations of } A_n = \{1 \cdots n\} \text{ is } n!$$

B.S. $P(1)$

I.H. $P(n)$

I.S. $P(n) \to P(n+1)$

Choosing from

Inserting into

DH 2.11: Generate All Permutations

Design an algorithm which, given a positive integer N, produces all the permutations of A_N.

DH 2.10: Permutation Checking

- An integer n
- An array of integers P of length n

To check whether P is a permutation of $1 \cdots n$?

DH 2.10: Permutation Checking

- An integer n
- An array of integers P of length n

To check whether P is a permutation of $1 \cdots n$?

DH 2.10: Permutation Checking

- An integer n
- An array of integers P of length n

To check whether P is a permutation of $1 \cdots n$?

- Boolean array $[1 \cdots n]$

DH 2.10: Permutation Checking

- An integer n
- An array of integers P of length n

To check whether P is a permutation of $1 \cdots n$?

- Boolean array $[1 \cdots n]$
- Sort

DH 2.10: Permutation Checking

- An integer n
- An array of integers P of length n

To check whether P is a permutation of $1 \cdots n$?

- Boolean array $[1 \cdots n]$
- Sort
- $\forall x : x \in [1 \cdots n]$
- check for duplication

Stackable Permutations

Definition (Stackable Permutations)

`read(X)`: >> X

`print(X)`: << X

`push(X, S)`: $S \Leftarrow X$

`pop(X, S)`: $X \Leftarrow S$

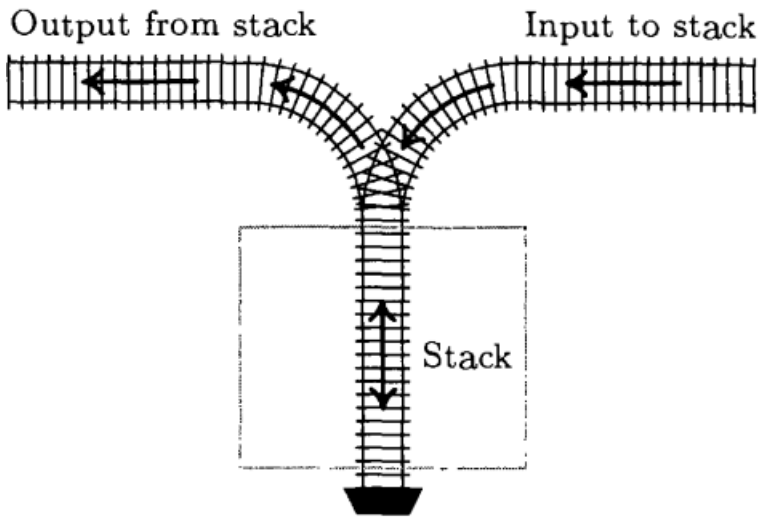
Definition (Stackable Permutations)

`read(X)`: >> X

`print(X)`: << X

`push(X, S)`: $S \Leftarrow X$

`pop(X, S)`: $X \Leftarrow S$



DH 2.12: Stackable Permutations

(a) Show that the following permutations can be obtained by a stack:

(i) $(3, 2, 1)$
(ii) $(3, 4, 2, 1)$
(iii) $(3, 5, 7, 6, 8, 4, 9, 2, 10, 1)$

DH 2.12: Stackable Permutations

(a) Show that the following permutations can be obtained by a stack:

(i) $(3, 2, 1)$
(ii) $(3, 4, 2, 1)$
(iii) $(3, 5, 7, 6, 8, 4, 9, 2, 10, 1)$

DH 2.13: Stackable Permutations Checking Algorithm

Design an algorithm that checks whether a given permutation can be obtained by a stack.

`push` `pop` `is-empty`

DH 2.12: Stackable Permutations

(b) Prove that the following permutations cannot be obtained by a stack:

(i) $(3, 1, 2)$
(ii) $(4, 5, 3, 7, 2, 1, 6)$

DH 2.12: Stackable Permutations

(c) How many permutations of A_4 *cannot* be obtained by a stack?

DH 2.12: One Stack and Permutations

(a) Sufficient and necessary condition

(b) How many?

Thank
You!