



Enumeration

李昕威

Sprout



影片 Q & A ?

Sprout



前言

< 算法班不是競賽培訓班 >

1. 如何枚舉一個集合的子集？
2. 如何枚舉一個重複集合的子集？
3. 如何求出一個元素皆相異的數列的所有排列？
4. 如何求出一個任意數列的所有排列？

如何有規律的枚舉情況、如何排序一個數列.....etc

甚麼是算法？

Sprout



枚舉子集合

Problem

給定一個集合 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設 S 中的元素皆相異。

請輸出所有 S 的子集合。

EX

$S = \{1, 2, 3\}$

輸出：

$\{\}$ 、 $\{1\}$ 、 $\{2\}$ 、 $\{3\}$ 、 $\{1, 2\}$ 、 $\{1, 3\}$ 、 $\{2, 3\}$ 、 $\{1, 2, 3\}$

Sprout



枚舉子集合

Problem

給定一個集合 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設 S 中的元素皆相異。

請輸出所有 S 的子集合。

如何枚舉？

分別考慮每個數字

可以取、或者不取

Sprout



枚舉子集合

使用遞迴進行實作

<https://ideone.com/fGRoYA>

Sprout



枚舉子集合

Problem

給定一個集合 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設 S 中的元素皆相異。

請輸出所有 S 的子集合。

Observation

每個數字取或不取，可以對應到 0 還有 1

Sprout



枚舉子集合

$$S = \{1, 2, 3\}$$

$$(000) \Rightarrow \{\}$$

$$(001) \Rightarrow \{3\}$$

$$(110) \Rightarrow \{1, 2\}$$

$$(111) \Rightarrow \{1, 2, 3\}$$

以此類推

Sprout



枚舉子集合

Problem

給定一個集合 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設 S 中的元素皆相異。

請輸出所有 S 的子集合。

因此，我們可以使用數字的二進位表示法來枚舉

EX

$S = \{1, 2, 3\}$ ，因為 $|S|=3$ ，因此我們需要

000、001、010、011、100、101、110、111，這些數字

Sprout



枚舉子集合

以位元來枚舉

<https://ideone.com/701Tx0>

Sprout



枚舉子集合

Problem

給定一個集合 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設 S 中的元素皆相異。

請輸出所有 S 的子集合。

複雜度分析

每個數字都可以取或不取，因此有 2^N 種選法

每種選法要檢查 N 個數字，因此為 $O(N * 2^N)$

Sprout



枚舉子集合

注意到右方的那個式子
可以發現我們所得到的上界
已經是非常緊的了

$$\sum_{i=0}^N i \binom{N}{i} = N \times 2^{N-1}$$

Sprout



枚舉子集合

Problem

給定一個集合 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
所有數字皆落在 $[1, m]$ 之間， S 中的元素「不一定」相異。
請輸出所有 S 的子集合。

EX

$S = \{1, 2, 2, 3\}$

$\{\}$ 、 $\{1\}$ 、 $\{2\}$ 、 $\{1, 2\}$ 、 $\{1, 3\}$ 、 $\{1, 2, 2\}$ 、 $\{2, 3\}$ 、
 $\{2, 2, 3\}$ 、 $\{2, 2\}$ 、 $\{1, 2, 2, 3\}$ 、 $\{3\}$ 、 $\{1, 2, 3\}$

Sprout



枚舉子集合

Problem

給定一個集合 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
所有數字皆落在 $[1, m]$ 之間， S 中的元素「不一定」相異。
請輸出所有 S 的子集合。

Observation

在上一個問題中，每個元素只有兩種情況，取或不取
那這一個問題呢？

Sprout



枚舉子集合

Problem

給定一個集合 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
所有數字皆落在 $[1, m]$ 之間， S 中的元素「不一定」相異。
請輸出所有 S 的子集合。

Observation

假設 a_i 出現了 x 次，那麼可以選 0 個、1 個、2 個... 或者是 x 個

Sprout



枚舉子集合

Problem

給定一個集合 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
所有數字皆落在 $[1, m]$ 之間， S 中的元素「不一定」相異。
請輸出所有 S 的子集合。

Observation

對於每個相同的數字，只要決定了要選幾個
那麼最後選出來的子集合就確定了

Sprout



枚舉子集合

使用遞迴進行實作
這樣寫的複雜度如何呢？

<https://ideone.com/Eg9Ppd>

Sprout



數列的排列

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設其中所有數字皆相異。

請輸出其所有不同的排列。

EX

$S = \{1, 2, 3\}$

(123)、(132)、(213)、(231)、(312)、(321)

Sprout



數列的排列

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設其中所有數字皆相異。

請輸出其所有不同的排列。

Obervation

先選一個數字放在第一位

再選一個數字放在第二位

重複上面的動作直到結束

Sprout



數列的排列

以DFS進行實作

<https://ideone.com/oT1L3S>

Sprout



數列的排列

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設其中所有數字皆相異。

請輸出其所有不同的排列。

複雜度分析

選第一個數字時有 N 種選法

選第二個數字時有 $N-1$ 種選法

選第 i 個數字時有 $N-i$ 種選法

Sprout



數列的排列

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設其中所有數字皆相異。

請輸出其所有不同的排列。

複雜度分析

總共的選法為 $N!$ ，因此複雜度即為 $O(N!)$

這樣的複雜度無法再改進了

Sprout



數列的排列

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設所有數字都在 $[1, m]$ 之間

請輸出其所有不同的排列。

EX

$S = \{1, 2, 2, 3\}$

(1223) 、 (3221) 、 (1232) 、 (3212) 、 (1322) 、 (3122) 、
(2123) 、 (2132) 、 (2213) 、 (2231) 、 (2312) 、 (2321)

Sprout



數列的排列

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。

假設所有數字都在 $[1, m]$ 之間

請輸出其所有不同的排列。

Observation

與所有數字皆相異的版本相比

在這個問題中，相同的數字有許多個

因此我們需要紀錄每個數字被選了幾次

Sprout



枚舉子集合

使用遞迴進行實作
這樣寫的複雜度如何呢？

<https://ideone.com/at6uLH>

Sprout



一些枚舉的題目

Sprout



前言

學會如何枚舉的話，就能夠解許多的問題了
只是解開的速度可能會很慢就是了

為了改善這樣的情況
我們有時候會搭配剪枝來進行枚舉
又或者我們會配合問題的性質，以較佳的方式來枚舉

Sprout



八皇后問題

Problem

給定一個西洋棋盤，要在上面放八個皇后。

皇后的攻擊範圍為上下左右與四個對角線。

請輸出一個放法使的任兩個皇后之間不會互相攻擊。

Observation

要對甚麼東西進行枚舉呢？

Sprout



八皇后問題

Problem

給定一個西洋棋盤，要在上面放八個皇后。

皇后的攻擊範圍為上下左右與四個對角線。

請輸出一個放法使的任兩個皇后之間不會互相攻擊。

Observation

每個棋盤格上，有兩個情況，放上皇后、或者是不放

因此可以枚舉每個棋盤上是否有皇后

另外還可以注意到，每一個橫排只需要枚舉一個皇后即可

Sprout



八皇后問題

複雜度如何呢？
有沒有更好的寫法？

<https://ideone.com/SAH0D9>

Sprout



騎士之旅

Problem

給定一個西洋棋盤，其中一個格子上放了一個騎士。

騎士每次可以移動到八個日字型的對角上。

請給出一種路徑使的騎士能夠剛好經過每一個格子一次。

Sprout



騎士之旅

Problem

給定一個西洋棋盤，其中一個格子上放了一個騎士。
騎士每次可以移動到八個日字型的對角上。
請給出一種路徑使的騎士能夠剛好經過每一個格子一次。

Oberservation

要枚舉甚麼東西呢？

Sprout



騎士之旅

Problem

給定一個西洋棋盤，其中一個格子上放了一個騎士。
騎士每次可以移動到八個日字型的對角上。
請給出一種路徑使的騎士能夠剛好經過每一個格子一次。

Observation

每次騎士移動時最多有八種選擇
因此我們可以枚舉騎士下一步的位置

Sprout



騎士之旅

Problem

給定一個西洋棋盤，其中一個格子上放了一個騎士。
騎士每次可以移動到八個日字型的對角上。
請給出一種路徑使的騎士能夠剛好經過每一個格子一次。

Observation

每次騎士移動時最多有八種選擇
因此我們可以枚舉騎士下一步的位置

Sprout



騎士之旅

好像有時候會跑得很慢
有沒有甚麼方法可以加速？

<https://ideone.com/KbiHch>

Sprout



騎士之旅

Problem

給定一個西洋棋盤，其中一個格子上放了一個騎士。
騎士每次可以移動到八個日字型的對角上。
請給出一種路徑使的騎士能夠剛好經過每一個格子一次。

Observation

如果騎士可以移動到的格子，都已經走過了
但卻有格子還沒走到時，代表那些格子沒被走到（當然）

Sprout



騎士之旅

Problem

給定一個西洋棋盤，其中一個格子上放了一個騎士。
騎士每次可以移動到八個日字型的對角上。
請給出一種路徑使的騎士能夠剛好經過每一個格子一次。

Observation

那些格子比較有可能「無法被走到」？

Sprout



騎士之旅

Problem

給定一個西洋棋盤，其中一個格子上放了一個騎士。
騎士每次可以移動到八個日字型的對角上。
請給出一種路徑使的騎士能夠剛好經過每一個格子一次。

Oberservation

如果一個格子，日字形的鄰居中，有越多格子被走過
那麼這個格子就會越難以被走到

Sprout



騎士之旅

Problem

給定一個西洋棋盤，其中一個格子上放了一個騎士。
騎士每次可以移動到八個日字型的對角上。
請給出一種路徑使的騎士能夠剛好經過每一個格子一次。

Oberservation

因此騎士就應該先將難以走到的點，先走過會比較好
因此，我們在枚舉騎士的移動時，要優先往「到那個格子後，還能走的路是最少的」的格子移動

Sprout



騎士之旅

實作就交給有興趣的人了

在這一題中，演示了透過問題的特性，來加速枚舉的方法。

Sprout



棋盤路徑

Problem

給定一個 $7*7$ 的棋盤。

移動時只能往上下左右四個方向。

請問有多少條路徑從左上角開始走到右下角，
並且剛好通過每一個格子一次？

Sprout



棋盤路徑

Problem

給定一個 $7*7$ 的棋盤。

移動時只能往上下左右四個方向。

請問有多少條路徑從左上角開始走到右下角，
並且剛好通過每一個格子一次？

Observation

可以枚舉每次移動的方向

Sprout



棋盤路徑

Problem

給定一個 $7*7$ 的棋盤。

移動時只能往上下左右四個方向。

請問有多少條路徑從左上角開始走到右下角，
並且剛好通過每一個格子一次？

Observation

可以枚舉每次移動的方向

只有這樣的話，超慢，共約760億次的遞迴

(by competitive Programming handbook)

Sprout



棋盤路徑

Problem

給定一個 $7*7$ 的棋盤。

移動時只能往上下左右四個方向。

請問有多少條路徑從左上角開始走到右下角，
並且剛好通過每一個格子一次？

Observation

觀察到，第一步往右走，和第一位往下走
兩種走法其實有個對稱的對應

Sprout



棋盤路徑

Problem

給定一個 $7*7$ 的棋盤。

移動時只能往上下左右四個方向。

請問有多少條路徑從左上角開始走到右下角，
並且剛好通過每一個格子一次？

Observation

因此可以限定第一步要往右走
共需要約380億次遞迴

Sprout



棋盤路徑

Problem

給定一個 $7*7$ 的棋盤。

移動時只能往上下左右四個方向。

請問有多少條路徑從左上角開始走到右下角，
並且剛好通過每一個格子一次？

Observation

如果不小心走到最右角了，但還沒走過所有格子
這時候就可以馬上return了

Sprout



棋盤路徑

Problem

給定一個 $7*7$ 的棋盤。

移動時只能往上下左右四個方向。

請問有多少條路徑從左上角開始走到右下角，
並且剛好通過每一個格子一次？

Observation

共需要約20億次遞迴

Sprout



棋盤路徑

Problem

給定一個 $7*7$ 的棋盤。

移動時只能往上下左右四個方向。

請問有多少條路徑從左上角開始走到右下角，
並且剛好通過每一個格子一次？

Observation

如果目前的路徑，將整個棋盤切成了兩個連通塊
則也可以馬上return

Sprout



棋盤路徑

Problem

給定一個 $7*7$ 的棋盤。

移動時只能往上下左右四個方向。

請問有多少條路徑從左上角開始走到右下角，
並且剛好通過每一個格子一次？

Observation

共需6900萬次遞迴

Sprout



棋盤路徑

實作略。

這類實作題適合拿來練習

有興趣的人請自行加油唄

在以上的問題中，我們見到了剪枝是如何在枚舉中發揮巨大功效的

Sprout



Permutation

Problem

給定一個數列，求其所有的排列。

實際上，C++有著相關的函式可以使用

`next_permutation`

`prev_permutation`

Sprout



Permutation

`next_permutation`

1. 得到數列的下一個字典序的排列
2. 如果新的排列的字典序，比原本的還小，回傳`false`；否則回傳`true`

ex

$(132) \Rightarrow (213)$ 回傳`true`

$(321) \Rightarrow (123)$ 回傳`false`

Sprout



Permutation

next_permutation

1. 也適用於有相同數字的情況
2. 每次操作的複雜度為均攤 $O(1)$

ex

$(1223) \Rightarrow (1232)$

均攤 $O(1)$ ，代表每次操作都是 $O(1)$ 嗎？

Sprout



二分搜

Sprout



單調性

EX

000001111122222

000001111111111

01234567889999999

Sprout



單調性

我們稱以下情況有單調性

1. 若 $x \leq y$ 可以推導到 $f(x) \leq f(y)$
2. 若 $x \leq y$ 可以推導到 $f(x) \geq f(y)$
3. 若 $x \leq y$ 可以推導到 $f(x) > f(y)$
4. 若 $x \leq y$ 可以推導到 $f(x) < f(y)$

Sprout



單調性

若一個問題有著單調性
則我們在枚舉答案時就能夠很漂亮的進行「剪枝」

EX

找出第一個出現的「1」：

00000000000000001111111111111111111111

Sprout



單調性

EX

找出第一個出現的「1」：

000000000000000000001111111111111111111111111

以人眼看的話，很清楚的就知道在哪裡

但如果以電腦、以陣列的角度來看呢？

如果這個序列長度為 N ，則一個一個檢查的複雜度為 $O(N)$

有沒有更好的方法呢？

Sprout



單調性

EX

找出第一個出現的「1」：

0000000 「0」 0000000001111111111111111111111111

隨便亂戳一個位置，發現是0

根據單調性，我們能夠知道這個位置的左邊都是0

因此那些位置都不可能是答案了，可以直接剪掉

Sprout



單調性

EX

找出第一個出現的「1」：

0000000 「0」 00000000011111111111111111111111

=> 「0」 00000000011111111111111111111111

隨便亂戳一個位置，發現是0

根據單調性，我們能夠知道這個位置的左邊都是0

因此那些位置都不可能是答案了，可以直接剪掉

Sprout



單調性

EX

找出第一個出現的「1」：

000000000011「1」111111111111111111

再隨便亂戳一個位置，發現是1

根據單調性，我們能夠知道這個位置的右邊都是1

因此那些位置都不可能是答案了，可以直接剪掉

Sprout



單調性

EX

找出第一個出現的「1」：

000000000011「1」11111111111111111111

=> 000000000011「1」

再隨便亂戳一個位置，發現是1

根據單調性，我們能夠知道這個位置的右邊都是1

因此那些位置都不可能是答案了，可以直接剪掉

Sprout



單調性

EX

找出第一個出現的「1」：

0000000000111

=> 000000000 「0」 111

=> 0111

=> 0 「1」 11

=> 01

Sprout



單調性

EX

找出第一個出現的「1」：

01

照我們上面的作法

這時候不管戳到哪個位置，都不會改變誼

所以這個方法其實是有問題的嗎？

Sprout



單調性

EX

找出第一個出現的「1」：

01

換個角度想，當我們發現數列的長度只剩下二的時候
就等於是找到了0與1之間的分界點
左邊的一定是0、右邊的一定是1
因此輸出右邊的那個位置就是答案了

Sprout



單調性

1. 隨便戳一個位置，找出那個位置的值
2. 根據單調性，可以將數列的某些部分剪掉
3. 直到最後只剩下兩個數字，找到分界點

要怎麼戳，效率才會最好？

Sprout



單調性

1. 隨便戳一個位置，找出那個位置的值
2. 根據單調性，可以將數列的某些部分剪掉
3. 直到最後只剩下兩個數字，找到分界點

要怎麼戳，效率才會最好？

每次戳中間的效率最好

Sprout



單調性

令 $T(N)$ 為：

給定長度為 N 、且具有單調性的01數列
找到第一個出現的1位置所需要的時間

則：

$$T(N) = \min_{i=1}^N (\max(T(i), T(N - i))) + O(1)$$

Sprout



單調性

可以發現， i 取 $N/2$ 的時候結果會最好

因此：

$$T(N) = \min_{i=1}^N (\max(T(i), T(N-i))) + O(1) = T(\lceil \frac{N}{2} \rceil) + O(1)$$

Sprout



單調性

再根據一些數學分析或者直覺
可以發現 $T(N) = O(\log(N))$
到此我們完成了複雜度的分析

而這個方法，也被我們稱呼為二分搜

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Observation

我們考慮 `insertion sort`

Sprout



二分搜

Selection sort

1. 從第一個數字開始，不斷加數字加入到新的序列中
2. 加入的方式是，找到這個新序列中第一個比其小的數字
3. 然後將其放到其後面

EX

$S = \{4312\}$

$\{4\} \Rightarrow \{34\} \Rightarrow \{134\} \Rightarrow \{1234\}$

Sprout



二分搜

Selection sort

1. 從第一個數字開始，不斷加數字加入到新的序列中
2. 加入的方式是，找到這個新序列中第一個比其小的數字
3. 然後將其放到其後面

複雜度分析

每次新加入一個數字的時候，需要花 $O(N)$ 的時間檢查
因此總共需要 $O(N^2)$ 的時間

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Observation

我們考慮 `insertion sort`
每次檢查時，真的需要 $O(N)$ 嗎？

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Observation

一個排序好的數列，有著單調性
若是 $x \leq y$ ，則 $f(x) \leq f(y)$

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Observation

在做 `selection sort` 的時候
等於是在維護一個排序好的數列

Sprout



二分搜

Selection sort

EX

$S = \{4312\}$

$\{4\} \Rightarrow \{34\} \Rightarrow \{134\} \Rightarrow \{1234\}$

每次加完數字後，所生成的數列都是排序好的

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Observation

因為排序好的數列有著單調性

因此當我們要加入新的一個數字時，就不需要一個一個檢查！

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Observation

因為排序好的數列有著單調性

我們可以利用二分搜，在 $O(\log N)$ 的時間內知道新數字要放在哪

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Observation

因為排序好的數列有著單調性

我們可以利用二分搜，在 $O(\log N)$ 的時間內知道新數字要放在哪

因此，如果我們也可以在 $O(\log N)$ 內插入的話

就能夠在 $O(N \log N)$ 的時間內排序完成

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Observation

有甚麼東西可以讓我們快速插入一個數字呢？

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Observation

有甚麼東西可以讓我們快速插入一個數字呢？
上一堂課所講的二元搜尋樹

Sprout



二分搜

Problem

給定一個數列 $S = \{a_0, a_1, \dots, a_{(N-1)}\}$ 。
請輸出其排序後的結果。

Selection sort, 進階版

1. 在二元搜尋樹上進行二分搜，找到要插入的位置
2. 將數字插入，並繼續處理下個數字直到結束

Sprout



小結

Sprout



Enumeration

對於打資訊競賽的人來說，枚舉可能是個很沒用的東西
礙於時間複雜度的緣故，因此許多場合都不會出枚舉的題目
真的是這樣嗎？

Sprout



Enumeration

二分搜可以看成是好的枚舉再加上好的剪枝

三分搜同上

DP是枚舉所有可能，同時紀錄最佳解來使自己不需要重複枚舉
一些DP優化是因為有著好的性質，所以能夠對轉移剪枝

GREEDY是問題有著好性質，每次枚舉都只需要考慮一種可能

二元搜尋樹是有著好的性質，因此在枚舉時可以進行大量剪枝

線段樹是藉由增加記錄的資料，來降低需要枚舉的東西

· · · etc.

Sprout



Q & A

Sprout