Exhaustion, branch & bound

Theo

Weapon

Recursion

Exhaustion

- Aka Brute Force
- Many "answers" (some of them are wrong!)
- Check all answers one by one

When to use?

- When the problem has small constraints
  e.g. 2<=N<=9
- When you have no idea...
  not every problem has a nice solution!
- When you want to get a non-zero score

Simple examples

- POJ 1248 Safecracker

Simple examples

- What will you do if you are solving it manually?
- e.g. target = 1, ALPHABET = “ABCDEFG”

Simple examples

- What will you do if you are solving it manually?
- e.g. target = 1, ALPHABET = “ABCDEFG”
- Try “ABCDE” sum = 1-2^2+3^3-4^4+5^5 = 2893
- not answer ☹

Simple examples

- What will you do if you are solving it manually?
- e.g. target = 1, ALPHABET = “ABCDEFG”
- Try “ABCDE” sum = 1-2^2+3^3-4^4+5^5 = 2893
- not answer ☹
- Try “ABCDF” sum = 1-2^2+3^3-4^4+6^5 = 7544
- not answer ☹

Simple examples

- What will you do if you are solving it manually?
- e.g. target = 1, ALPHABET = “ABCDEFG”
- Try “ABCDE” sum = 1-2^2+3^3-4^4+5^5 = 2893
- not answer ☹
- Try “ABCDF” sum = 1-2^2+3^3-4^4+6^5 = 7544
- not answer ☹
- Try “GABDC” sum = 7-1^2+2^3-4^4+3^5 = 1
- answer ☺

Simple examples

- How fast can the exhaustion go?
- Max number of possible answers = 26*25*24*23*22
- = 6375600, which is quite small for computers

Simple examples

- HKOI 2000 Magic triangle

Simple examples

- You have a triangle with N (1 <= N <= 5) layers.
- The i-th layer has i node.
- Every node is filled with an integer from 1 to N(N+1)/2 exactly once
- Number at top = the absoulte difference of the two numbers directly under it.
- The triangle is partially-filled. You are asked to finish the triangle.

Simple examples

- Similiar idea
- For every unfilled node, try to fill one number from possible candidates.
- Check if the config is correct.

Simple examples

- How fast can the program run?
- Worse case: every node is unfilled
  number of possible configs:P*(P-1)*(P-2)...*1 = P!, where P = N(N+1)/2 = number of empty slots
- N=3, P! = 720 fast
- N=4, P! = 3628800 okay
- N=5, P! = 1307674368000 TLE

Simple examples

- Optimization
- Actually, we don’t need to exhaust all numbers
- Why?

Simple examples

- Exhaust all numbers on bottom layers only
- Generate all numbers on top of the bottom layer
- Check if such generation is possible

Simple examples

- How fast can the program run?
- Worse case: every node is unfilled
  - number of possible configs: (15 P 5) ~= 15^5
- N=5, 15^5 <= 5*10^8 no problem at all

Branch and Bound

- The searching process can be viewed as a rooted tree
- For some of the branch, they may look “useless” to search

  if we can remove them (in program), our program can try fewer configs-->faster

More examples

- Given a grid of NxN (1<=N<=6), count number of paths such that

  the path always start from top left corner and end at bottom left corner

  each cell is visited exactly once

More examples

- Easiest: calculate by hand!
- May make mistake
- You may fall asleep

    N=6 ans=1770

More examples

- Simple exhaustion
- Try to walk from top left corner, record the cells that was visited
- If in some cases we cannot walk furthurmore, return
- If in some cases we are in bottom left corner and we visited all cells, add one to answer

More examples

- How fast can the program run?
- Rough estimation:

  start: 1 possible config

  for each step we have an average of 2.5 direction to go

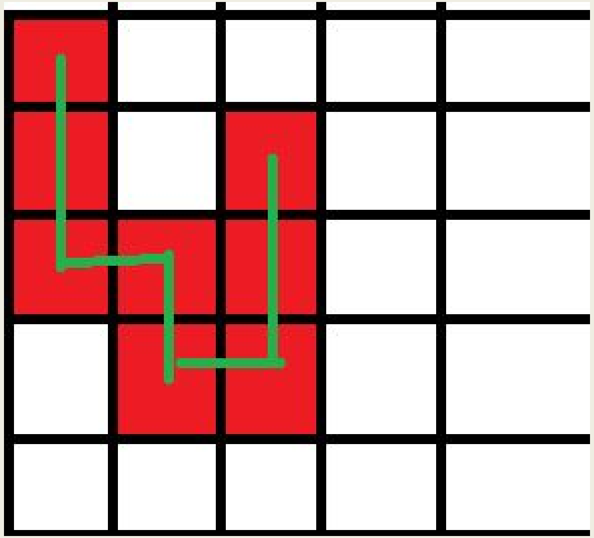
  number of steps required ~ N^2

  number of config ~ 2.5^(N^2)

  N=6, 2.5^(N^2) ~ 2*10^14 TLE

More examples

- Optimization?

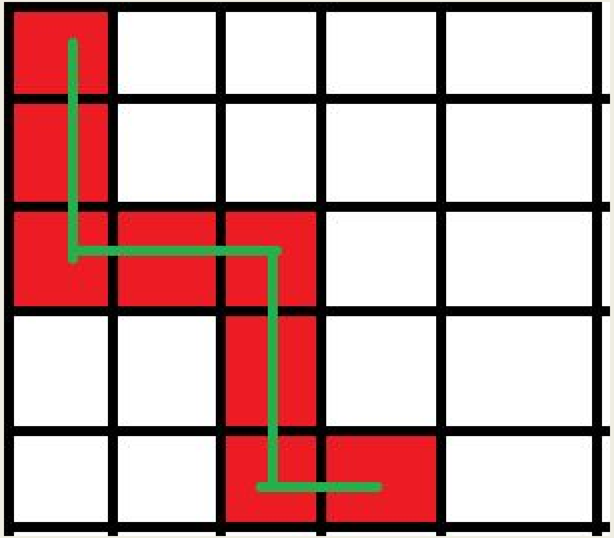


More examples

- For every unvisited cell, except the cells that is adjacent to the current cell we are in and the bottom-left cell, there should be at least two unvisited cells adjacent to it.
- Why?
- if some cells contradicts, we can say there is no sol for this config.

More examples

- More optimization?



More examples

- If we walk to a point such that two disconnected components are formed, no solution exists
- Return

More examples

- Check disconnected components --> BFS/DFS, will be taught in Graph(I)
- Or you may ask us(trainers) afterwards

More examples

- Check disconnected components --> BFS/DFS, will be taught in Graph(I)
- Or you may ask us(trainers) afterwards

More examples

- HKOJ Chocolate

More examples

- Simple exhaustion:
  for every length p
  try to form first stick of length p
  try to add stick 1 to first stick
  try to add stick 1+2 to first stick
  ...
  try to add stick 1+3 to first stick
  ...
- Can we do better?

More examples

- Simple exhaustion:
  for every length p
      try to form first stick of length p
          try to add stick 1 to first stick
              try to add stick 1+2 to first stick
                  ...
              try to add stick 1+3 to first stick
                  ...

More examples

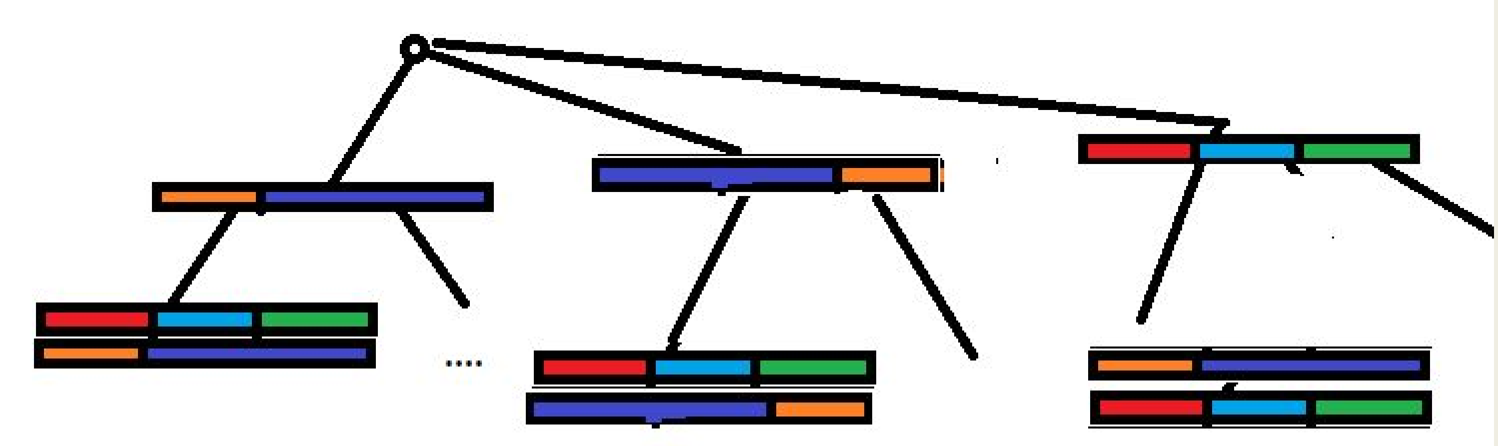
- e.g.
- 7
- 4 3 3 3 2 2 1
- 5,7 cannot be answer. why?

More examples

- e.g.
- 7
- 4 3 3 3 2 2 1
- 5,7 cannot be answer. why?

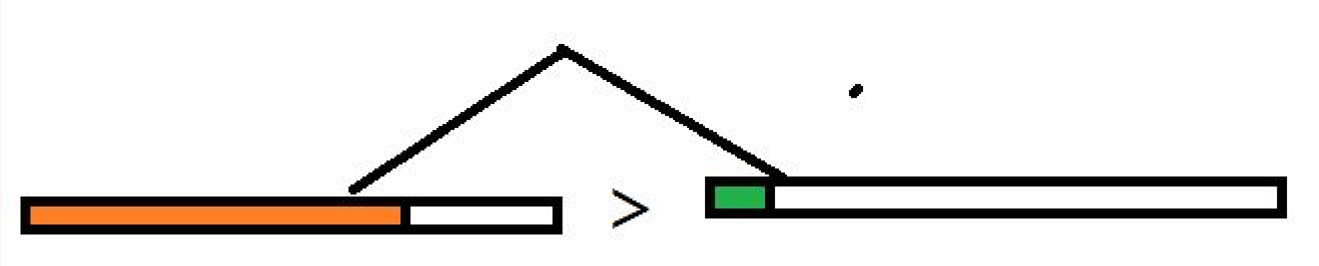
More examples

- Many repeated search!



More examples

- Sort the sticks in descending order first.
  why putting larger sticks first is better?

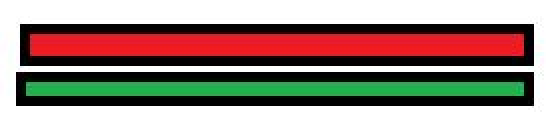


More examples

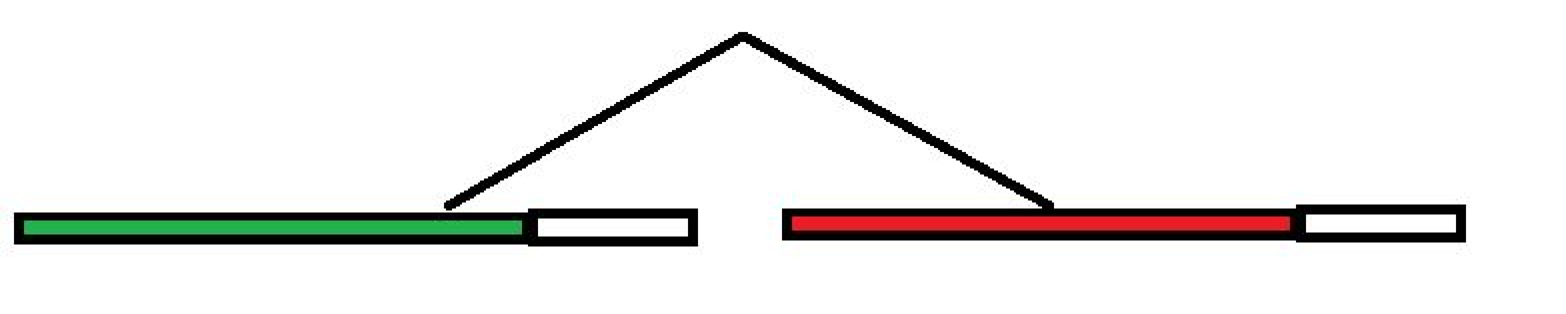
- Sort the sticks in descending order first.
- When we start forming stick i, we always put the largest stick we got.

More examples

- Suppose we have two sticks of same length



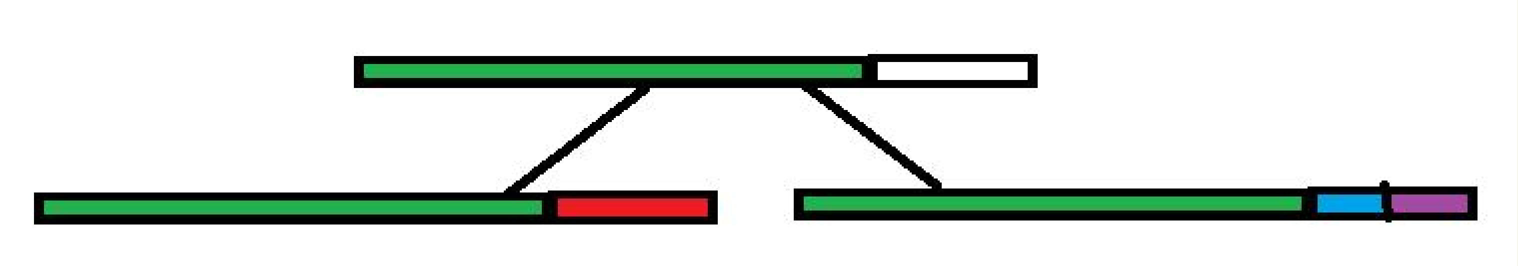
- Still repeated search!



More examples

- If we found that we cannot form ith stick using the stick of length p, we skip to next stick of length <p

More examples



More examples

- If we found that we cannot form ith stick using the stick of length p, we skip to next stick of length <p

Last example

- POJ 1190 生日蛋糕

Last example

- Search: attempt(V, A, L, LR, LH) : I know that if we still need to form V pi cubes of cake, I am in layer L, layer (L+1) has radius LR and height LH, the smallest area needed is A.
- Search is done when V=L=0
- Optimization?

Last example

- Search: attempt(V, A, L, LR, LH) : I know that if we still need to form V pi cubes of cake, I am in layer L, layer (L+1) has radius LR and height LH, the smallest area needed is A.
- Search is done when V=L=0
- Optimization?

Last example

- Lets say, in some point, least total volume of cake from layer 1 to layer i is > V, we can stop searching.
- Lets say, in some point, we know that current best answer < A, we can stop searching.
- Actually we can do better!

Last example

- Lets say, in some point, we know that current best answer < A + least area needed for layer 1..i, we can stop searching.
- How can we judge the least area needed and the least volume needed? (Remember that all radius and heights are integers!)
- Others are left as exercise

Exercise

- HKOJ
  1049
  1050
  2005
  2065
  3031
  (actually all copied from 2012's material)

Kimagure Cleaner

References

- 搜索與剪枝http://wenku.baidu.com/view/642854f80242a8956bece402.html
  A* 搜尋演算法
- http://zh.wikipedia.org/wiki/A*%E6%90%9C%E5%AF%BB%E7%AE%97%E6%B3%95

The End

Thanks!