GB21802 - Programming Challenges

Week 2 - Problem Solving Paradigms (Search)

Claus Aranha
caranha@cs.tsukuba.ac.jp

College of Information Science

2015-04-22,25

Last updated April 30, 2016

Last Week Results

Special Notes

None this time!

What is Search?

What is search in a common sense of the word (finding your cellphone in your home)

What is Search? (In a programming challenge)

- a way to think about the problem and the solution
- There are answer. We can classify the answers into correct and incorrect - We can also, usually, order the answers (both correct and incorrect)

Now you are thinking with search spaces

Thinking of problems as search problems allows us to define some strategies about how to analyse each of the possible searches.
Main question: In what order should we analyse the possible answers?

Search Paradigm

Complete Search/Brute Force Divide and Conquer Greedy Approach Dynamic Programming (Next week!)

Search Space/Search Structure

Most problems in programming challenges can be defined as search problems
But sometimes that is not the optimal way.
And sometimes more than one definition is possible!

Theoretical Example

Solving it with Complete (brute) search

11565 – Three layered solution

```
bool sol = false; int x,y,z;
for (x = -100; x <= 100 && !sol; x++)
  for (y = -100; y <= 100 && !sol; y++)
    for (z = -100; z <= 100 && !sol; z++)
      if (y != x && z != x && z != y &&
          x + y + z == A && x * y * z == B && x*x + y*y + z*z =
            if (!sol) printf("%d %d %d\n", x,y,z);
            sol = true;
          }
```

Divide and Conquer

D&C is a problem-solving paradigm in which a problem is made simpler by 'dividing' it into smaller parts.

- Divide the original problem into sub-problems;
- Find (sub)-solutions for each sub-problems;
- Combine sub-solutions to get a complete solution;

Examples

Quick Sort, Binary Search, etc...

Greedy

Greedy Example 1

Coin Change:
Given a target value V and a list of coin sizes, what is the minimum number of coins that we must use to represent V?
Example: V = 42, Coins = 25, 10, 5, 1 (a One coin means we can always make any value)
However, if V = 6, and coins = 4,3,1, the greedy algorithm does not reach an optimal solution.

Greedy Example 2

Load balancing: (UVA 410)
You have C chambers, and S < 2C specimens. You need to decide where each specimen should go to minimize "imbalance".
Insights:
- A chamber with 1 individual is always better than a chamber with 0 individuals.
- Order of chambers does not matter.
Greedy algorithm: Order the individuals by weight, and put one in each chambers until the chambers are full, then add one in each chamber backwards.

Greedy Example 3 - Dragon of LooWater (11292)

List of knights (with heights) and dragons (with diameter). Find the minimal height of knights that cut the heads of all dragons (knights can only cut the heads of dragons when D <= H).
Sort knights and dragons, and from smallest to biggest dragon, match with the smallest knight possible to solve.