GB21802 - Programming Challenges

Week 1 - Basic Problem Solving

Claus Aranha
caranha@cs.tsukuba.ac.jp

College of Information Science

2015-04-13

Last updated April 11, 2015

Some Notes Before the Class

Please check your “Programming Challenge” username!

Some people submitted me invalid usernames:

- “Oda”

Everyone else, please don’t forget to send your usernames!

Early submissions

We already had some early submissions – fantastic! Any problems or questions regarding the submission proccess?

Summary for Last Class

How the course works

- Monday Class: Theme exposition;
- Friday Class: Example problem and Q&A;
- Problems: Solve 4 problems every week;

How to submit the problems

- Make an account at `www.programming-challenges.com`
- Send your username to the professor (manaba or e-mail)
- Write the program in C, C++, Java or Pascal

How evaluation works

- Grade = number of problems submitted
- Try to submit one problem per week
- Comments and Participation counts

Summary for This Class

General Problem Solving :
A very important skill which is hard to teach formally;

Data Structures and Programming Challenges :
How to think of data structures outside of the classroom;

Problem Discussion :
Let's introduce last week's problems;

Relax, and ask questions!

No topic here is really new. Listen carefully!

Ask questions any time!

What is problem solving skill?

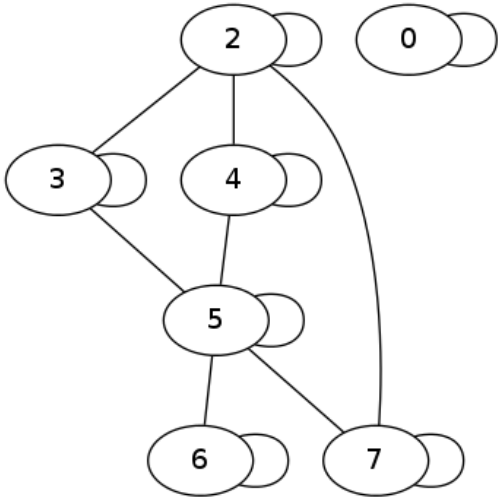
Steps to solve a problem

Don't know where to begin? Don't panic, keep calm, and try to follow these steps.

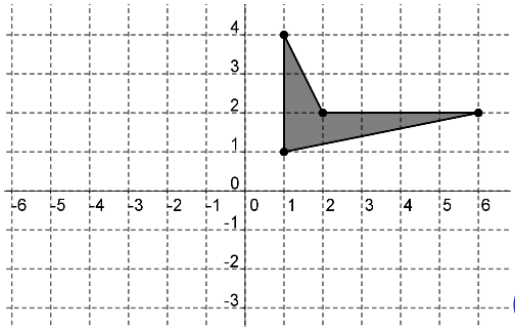
1. Read the input and output
2. Summarize the problem
3. Check for traps
4. Write the program
5. Test/Debug
6. Submit!

Problem Traps

Programming Challenges are (in)famous for including traps or "gotcha's" in their design.



Graphs: Connected? Directed? Redundant Edges? Negative weights?



Geometry: Overlapping? Concave? Negative coordinates? Collinear?

- Maximum number of entries:

NOW you can start to code

You should have done all of the previous steps in writing only!
Programming distracts from understanding the problem.

Steps for writing the solution

- Write the input/output first
- Make the program
- Release often philosophy
- Testing/Debugging

Speed and Memory Limits

Defining speed and memory limits
Algorithmic efficiency, Memory Efficiency, Programmer efficiency

Let's apply these steps to a simple problem

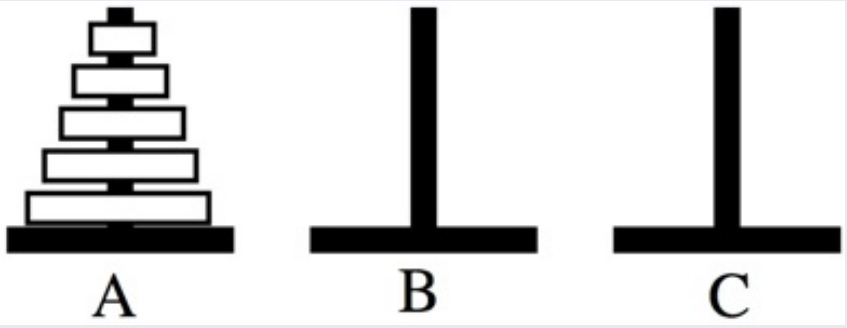
Data Structures

Data Structures

Data structures are the heart of a program

- Using the correct data type can make a problem much easier;
- Using the incorrect data type can make a problem much harder;

The towers of Hanoi



QUIZ: How do you represent the data in this problem?

An easy way to visualize the Towers of Hanoi

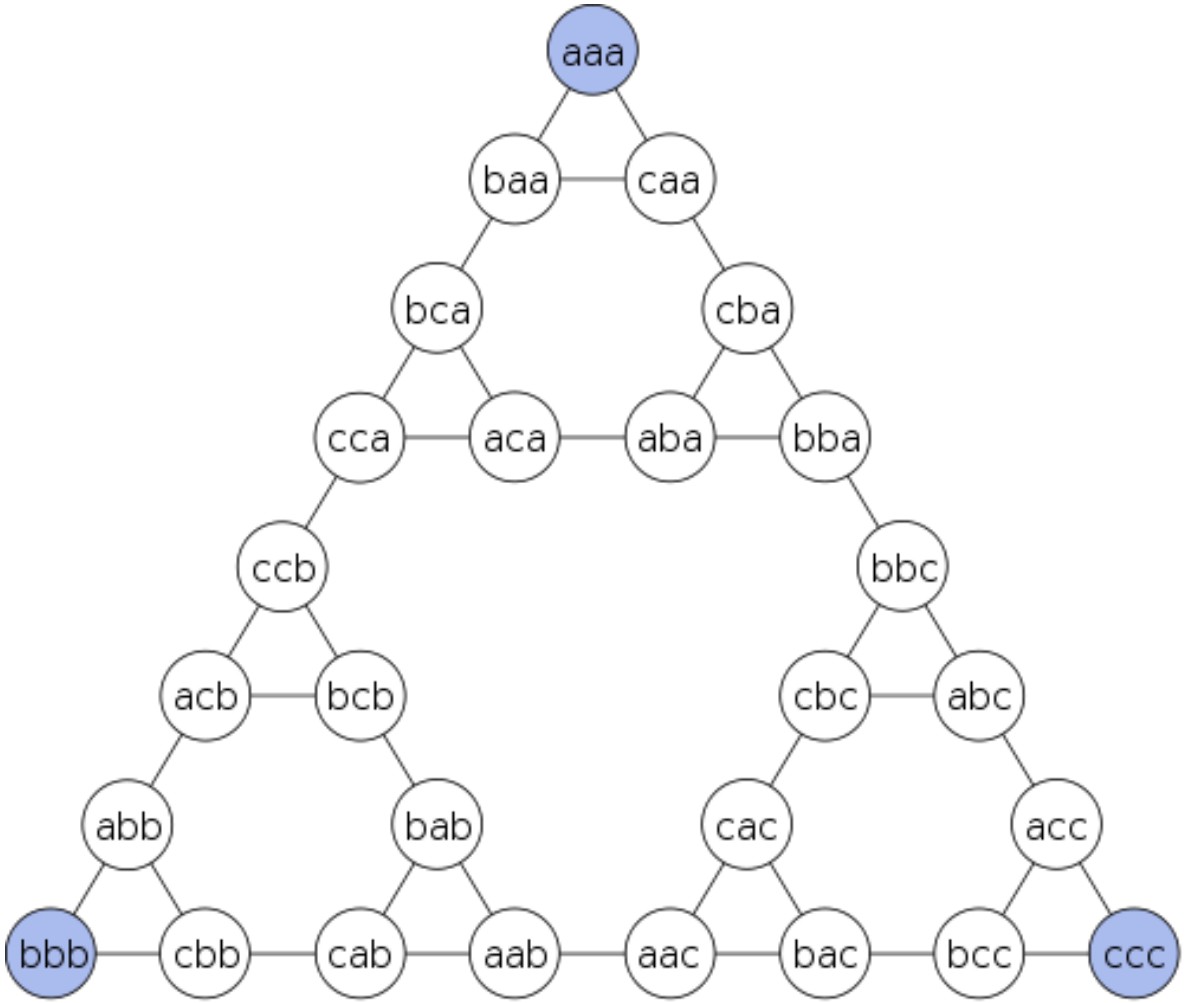
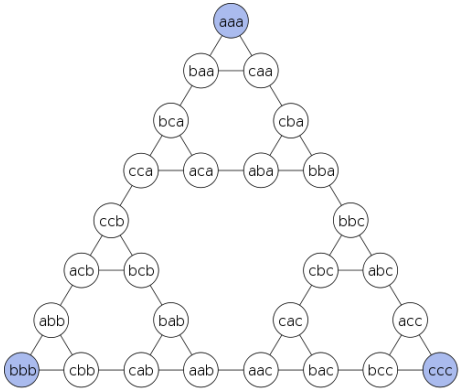


Image created by nonenmac

Explaining the Tower of Hanoi Data Structure

- Each node identifies an state in the problem;
- Each character in the string represents one disk and its position;
- We can have at most 3 state transitions at each state (can you prove it?)
- To solve the Towers of Hanoi problem, we find the path between the start and end states.
- (just beware of state explosion)



Know your data structures!

Let's talk about libraries

This week's problems

List of Problems

- The 3n+1 Problem
- Check the Check
- Erdos Numbers
- Contest Scoreboard

Let's give a quick look on each problem

For the rest of the week:

- Next class: Bring your solutions and questions!
- Submission deadline is 04-19 23:59:59 (Sunday)
- Have a nice week!

Welcome to Friday Class!

Current Solving Stats

- The 3n+1 Problem – Solved:
- Check the Check – Solved:
- Erdos Numbers – Solved:
- Contest Scoreboard – Solved:

Question and hint time!

- The 3n+1 Problem

Question and hint time!

- Check the Check

Question and hint time!

- Erdos Numbers

Question and hint time!

- Contest Scoreboard

Let's solve some different problems