

Practice for a Programming Contest

S. Mohsen Amiri

Outline

- Get Your Problems Solved
- Build Your Own Swiss Army Knife
- Know Better Your Swiss Army Knife: clean, simple, the tool you want to take everywhere.

Get Your Problems Solved

- Choose the best problem
 - Read problems carefully
 - Monitor score-list

- Solve the problems
 - Develop a **team-strategy** for the competition
 - Don't waste your time on a **misunderstood** problems
 - schedule for your **resources**
 - Make decisions on time

Get Your Problems Solved

- **Oops, It didn't work!**
 - Again, make decisions on time.
 - Especially hard ones!
 - Build an efficient protocol for fixing the problem
 - Detect the bug efficiently
 - Bugs in the algorithm
 - Bugs in the implementation
 - Solve the bug efficiently

Get Your Problems Solved

- Some links
 - <http://goo.gl/KGSJA>
 - <http://www.acmsolver.org/>
 - <http://uva.onlinejudge.org/>
 - [Art of Programming Contest](#)
 - Topcoder and its [tutorials](#)

Build Your Own Swiss Army Knife

- Read the [rules!](#)
 - for example the official compiler for C++ is **g++ 4.6.1, C++0x Mode**
- Handling Input and Outputs
- Ready to implement Algorithms and data structures in your favorite language

Build Your Own Swiss Army Knife

- Read the [rules!](#)
 - personal experience
 - `std::vector<int<int>> myVec2D;`

Build Your Own Swiss Army Knife

Handling Input and Outputs

- **Save your precious time**
- Heavily depend on your programming language
- Examples
 - <http://www.ugrad.cs.ubc.ca/~cs490/sec202/notes/intro/io-containers.pdf>
 - http://www.cs.ubc.ca/~simonsyd/acm/tutorial_io.html

Build Your Own Swiss Army Knife

- Ready to implement Algorithms and data structures in your favorite language
 - Will it work?
 - Can I code it?
 - Can I code it? or Do I really need to code it?
 - Can I debug it?
 - Is it fast enough?
 - Is it cool? or does it look geeky?

Build Your Own Swiss Army Knife

- Ready to implement Algorithms and data structures in your favorite language
 - Array
 - `std::vector`
 - Linked list, stack, queue
 - `std::list`, `std::stack`, `std::queue`
 - Binary search tree
 - `std::map`, `std::set`, `std::multimap`, `std::set`
 - Priority queue
 - `std::priority_queue` (max-heap)
 - Hash table (C++11, you can use them)
 -

Know Better Your Swiss Army Knife

- Swiss Army Knife: clean, simple, the tool you want to take everywhere.