ICPC Sessions
OR
How to Solve Problems

Sebastian Claici
sebastianclaici@gmail.com

November 28, 2012

What is the ACM ICPC?

What is the ACM ICPC?

- The ACM International Collegiate Programming Contest!

What is the ACM ICPC?

- The ACM International Collegiate Programming Contest!
- Team based (teams of 3)

What is the ACM ICPC?

- The ACM International Collegiate Programming Contest!
- Team based (teams of 3)
- The most prestigious global programming competition (since 1977)!

This is how it goes...

This is how it goes...

- Regionals and Finals

This is how it goes...

- Regionals and Finals
- We are part of the Northwestern European region

This is how it goes...

- Regionals and Finals
- We are part of the Northwestern European region
- Top 3 teams will qualify for the Finals in St. Petersburg

This is how it goes...

- Regionals and Finals
- We are part of the Northwestern European region
- Top 3 teams will qualify for the Finals in St. Petersburg
- This means we will make Germany, Belgium, the Netherlands and others cry!

What is this about?

What is this about?

- Algorithms; more than you did in previous years (if any)!

What is this about?

- Algorithms; more than you did in previous years (if any)!
- Actually using algorithms to solve problems!

What is this about?

- Algorithms; more than you did in previous years (if any)!
- Actually using algorithms to solve problems!
- Beating Cambridge (and everyone else)!

How to Prepare

How to Prepare

- Beginners:

How to Prepare

- Beginners:
    - USACO, UVa, Project Euler (latter not recommended)

How to Prepare

- Beginners:
    - USACO, UVa, Project Euler (latter not recommended)
- Intermediate:

How to Prepare

- Beginners:
  - USACO, UVa, Project Euler (latter not recommended)
- Intermediate:
  - http://poj.org, http://acm.tju.ed.cn, http://acm.sgu.ru, http://acm.timus.ru

How to Prepare

- Beginners:
  - USACO, UVa, Project Euler (latter not recommended)
- Intermediate:
  - http://poj.org, http://acm.tju.ed.cn, http://acm.sgu.ru, http://acm.timus.ru
- Everyone:

How to Prepare

- Beginners:
  - USACO, UVa, Project Euler (latter not recommended)
- Intermediate:
  - http://poj.org, http://acm.tju.ed.cn, http://acm.sgu.ru, http://acm.timus.ru
- Everyone:
  - Topcoder and Codeforces - make accounts on both

How to Prepare

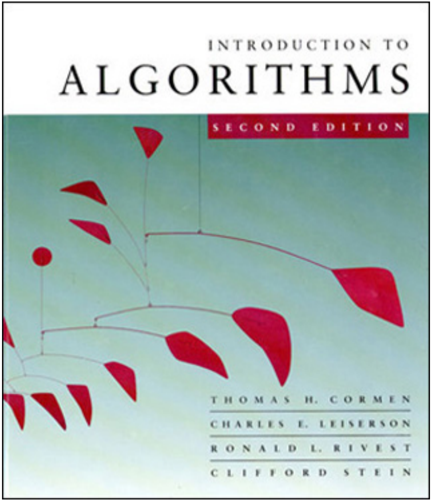
- Beginners:
  - USACO, UVa, Project Euler (latter not recommended)
- Intermediate:
  - `http://poj.org`, `http://acm.tju.ed.cn`, `http://acm.sgu.ru`, `http://acm.timus.ru`
- Everyone:
  - Topcoder and Codeforces - make accounts on both
- Romanians:

How to Prepare

- Beginners:
  - USACO, UVa, Project Euler (latter not recommended)
- Intermediate:
  - http://poj.org, http://acm.tju.ed.cn, http://acm.sgu.ru, http://acm.timus.ru
- Everyone:
  - Topcoder and Codeforces - make accounts on both
- Romanians:
  - Infoarena

Recommended Book(s)

- **Introduction to Algorithms**
  - Thomas Cormen, Charles Leiserson, Ronald Rivest, Clifford Stein
- **Algorithms in C/C++/Java/**
  - Robert Sedgewick



How about this: 90% of programmers, given 2 hours, the high-level language of their choice (including pseudocode), and a description of binary search could not implement it correctly.

How about this: 90% of programmers, given 2 hours, the high-level language of their choice (including pseudocode), and a description of binary search could not implement it correctly.

Can you?

Binary Search

```c
int binary_search(int *array, int n, int x)
{
  int lo = 0, hi = n - 1;
  // what most people get wrong
  while (lo < hi) {
    int mid = lo + (hi - lo) / 2;
    if (array[mid] < x)
      lo = mid + 1;
    else hi = mid;
  }

  if (lo == hi && array[lo] == x)
    return lo;
  return -1;
}
```

Ad-hoc Problems

Characteristics

Characteristics

- Most of them are very easy; some of them are very very hard

Characteristics

- Most of them are very easy; some of them are very very hard
- Don't require any special knowledge of algorithms

Characteristics

- Most of them are very easy; some of them are very very hard
- Don't require any special knowledge of algorithms
- There is always at least one in competitions

Strategies to solve

Strategies to solve

- Most of these problems are straightforward.

Strategies to solve

- Most of these problems are straightforward.
- However, most ad-hoc problems require careful reading and carefully sequencing the instructions given in the problem is usually enough to solve them.

Strategies to solve

- Most of these problems are straightforward.
- However, most ad-hoc problems require careful reading and carefully sequencing the instructions given in the problem is usually enough to solve them.
- Some require reasonable optimisations, and some degree of analysis to prune unnecessary steps.

Strategies to solve

- Most of these problems are straightforward.
- However, most ad-hoc problems require careful reading and carefully sequencing the instructions given in the problem is usually enough to solve them.
- Some require reasonable optimisations, and some degree of analysis to prune unnecessary steps.
- If it's not obvious, then there's only one piece of advice I can give you:

Don't Panic!