Course Introduction

Jingbo Shang and Mattox Beckman

University of Illinois at Urbana-Champaign
Department of Computer Science
Table of Contents

Introduction and Logistics

Objectives

Administrative Items
Welcome to CS 491 CAP!

Topics for discussion:
- Introduction
- How to practice
- Assignments
- Grading Policy
- Examples
- Practice Resources
Table of Contents

Introduction and Logistics

Objectives

Administrative Items
Meeting Time  Friday 14:00 in 1320 DCL

Instructors  Jingbo Shang (shang7) and Mattox Beckman (mattox)

Lecturers  Uttam, Victor, and Zhengkai

Syllabus  TBA
SIG ICPC Team

- Preparing for 2016 Mid-Central ICPC Regionals
  - Will discuss and collaboratively solve problems from this seminar’s problem sets
- Meeting Times TBD
- Chairs: Someone
- Officers: Others
- Coach: Jingbo Shang (shang7)
- Assistant Coach: Mattox Beckman (mattox)
- Mailing list:
  - Join us!
  - [https://www-s.acm.illinois.edu/cgi-bin/mailman/listinfo/icpc-l](https://www-s.acm.illinois.edu/cgi-bin/mailman/listinfo/icpc-l)
Prerequisites

- Proficiency in programming C, C++, or Java
  - CS 125 or equivalent
- Familiarity with basic data structures ideal, but not necessary (CS 225).
- Most important: eagerness to learn and practice!!
Why Compete in Programming Contents?

▶ It’s fun!

▶ Opportunity to learn:
  ▶ useful data structures, algorithms, and mathematical insights;
  ▶ practical applications of data structures and algorithms;
  ▶ how to code and debug effectively; and
  ▶ how to work well on a team.

▶ You’ll do really well on job interviews!
Programming Contests

- UIUC ICPC tryouts and practice
- ACM ICPC
  - Mid-central Regionals
  - World Finals
- Online contests
  - TopCoder SRMs, Codeforces
  - Facebook Hacker Cup
  - Google Code Jam
  - TopCoder Open
  - ... and many others ...
Online Judges

- Real contest problems
- Immediate Feedback
- Can emulate contest environment
- List of online judges:
  - Peking Online Judge http://poj.org
  - UVa Online Judge https://uva.onlinejudge.org/
  - ACM ICPC Live Archive https://icpcarchive.ecs.baylor.edu/
  - Sphere Online Judge (SPOJ): http://www.spoj.com/
  - Open Kattis https://open.kattis.com/
  - Saratov State Online Judge: http://acm.sgu.ru/

- **Get an account on each of these!** We will send you a link to collect your online judge IDs later.
Online Contests

- Occur 3–4 times per month.
- Top Coder Single Round Matches (SRMs).
  https://www.topcoder.com/
- Code Forces
  http://codeforces.com/
UIUC ICPC Team Meetings

- SIG ICPC Website:
  http://icpc.cs.illinois.edu/ipl.html
  - Contains announcements, practice summaries, and practice resources.

- Meeting Calendar:
  http://icpc.cs.illinois.edu/calendar.html

- Tryouts
  - September 23
  - October 7. Afternoon 11:00–16:00

- Practice contests on subsequent Saturdays.

- Details on http://icpc.cs.illinois.edu/calendar.html
Assignments

▶ One problem set per week, assigned at end of class.
  ▶ Problems will be rated by difficult with a point value
  ▶ Problems should be submitted on corresponding online judge.
▶ Completion of a problem set involves solving 4 points worth of problems.
  ▶ If you took CS 491 CAP last year, then you may not use 1 or 2 point problems towards your completion!
▶ Due within two weeks of assignment.
▶ **No Extensions!**

**NB:** Please do not copy-paste code from other sources. You are only hurting yourself if you do!
Grading Policy

- This class is Pass/Fail.
- You must complete 10 of 13 problem sets to pass.
- You may substitute two tryouts or practice contests for problem sets.
- Grades will be released through git. More details to come.
Approach to Solving ICPC Problems

1. **Read the problem statement carefully!**
   - Pay attention to the input/output format specification.
2. Abstract the problem.
3. Design an algorithm.
4. Implement and debug.
5. Submit.
6. AC!
   - (else GO TO 4... or maybe even 3)
Example Problem

- POJ 1000: A + B Problem
  - Input: two space separated integers, $a$ and $b$.
  - Constraints: $0 \leq a, b \leq 10$.
  - Output: $a + b$
```c
#include <stdio.h>

int main() {
    int a, b;
    scanf("%d %d", &a, &b);
    printf("%d\n", a + b);
    return 0;
}
```
Java Code for POJ 1000

```java
import java.io.*;
import java.util.*;

public class Main {
    public static void main(String args[]) throws Exception{
        Scanner cin=new Scanner(System.in);
        int a=cin.nextInt(), b=cin.nextInt();
        System.out.println(a+b);
    }
}
```
Example Problem

- POJ 1004 — Financial Management
  - Input: 12 floating-point numbers, each on a separate line
  - Output: Average of the numbers, rounded to two decimal places
  - Note that the answer must be preceded by a dollar sign ($)!
#include<stdio.h>

int main() {
    double sum = 0, buf;
    for(int i = 0; i < 12; i++) {
        scanf("%f", &buf);
        sum += buf;
    }
    printf("$%.2f\n", sum / 12.0);
    return 0;
}
Java Code for POJ 1004

```java
import java.util.*;

class Main {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        double d = 0;
        for (int i = 0; i < 12; ++i) {
            d += in.nextDouble();
        }
        System.out.printf("%.2f\n", d/12.0);
    }
}
```
Notes about POJ

- **Read the FAQs!** [http://poj.org/faq.htm](http://poj.org/faq.htm)
- Be aware of the compilers used by POJ and the other online judges.
- Write and test your code on your local machine, not in the online judge submission box.
- Use the best free IDE you like.
- When using `printf` format strings, use `%f`, instead of `%lf`. 
Questions?
Course Resources

- Course Website: https://pages.github-dev.cs.illinois.edu/sig-icpc/cs491-cap/
- Mailing list: https://www-s.acm.illinois.edu/cgi-bin/mailman/listinfo/icpc-l
- Piazza page: (NO solution posts!) https://piazza.com/class#fall2017/cs491cap
- UIUC ICPC team website: http://icpc.cs.illinois.edu/
- Announcements will be sent to the ICPC mailing list and put on Piazza
- Course materials will be available on the website