

CompSci 149s

If you are expecting a different class, you are in the wrong room. Also probably the wrong time. How many classes happen at 7PM on a Monday?

Q1: How many of you have done
competitive programming
before?

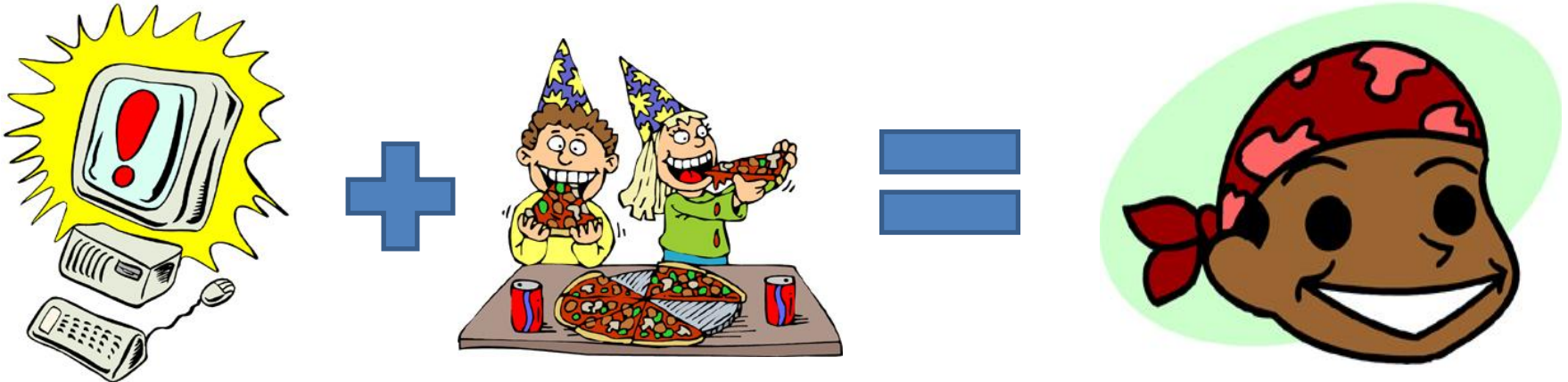
The Plan

1. Course Structure
2. Grades
3. A sample problem!
4. Greed! (if we have time)

Q2: How many of you plan to take this course for credit?

1. I am definitely taking this course for credit
2. I'm not sure...I might take the course for credit if it doesn't seem too painful. Or I might audit.
3. I'm definitely auditing.
4. I'm not sure if this competitive programming thing is for me.

My Recommendations



- If the idea of solving programming problems sounds fun and you're comfortable writing code on your own, I think you'll enjoy taking this course
- If this is a) your first time or b) you are serious about improving your programming skills, I think you'll enjoy taking this course for credit

A Normal Week

1. Working problems each week (with help from books, internet, instructors, friends). Maybe attending a discussion section.
2. Coming to class where we'll talk (briefly) about some approaches and do activities
3. Scrimmages, where we practice under competition-like conditions (recommended but not required)

How Will We Be Graded?

- If you put forth a reasonable effort, you will get an A- in the course
- A reasonable effort:
 - Attend class
 - Submit as many problems as you can per week (at least 2)
 - Attend the ACM-ICPC Regionals (Nov. 5th). This is REQUIRED.
 - Do the final post-Regionals project
- READ THE COURSE GUIDE FOR SPECIFICS

Getting Help

- Get all the help you need! From the internet, from textbooks, from your friends, from me outside of class, from your TAs

Submitting Problems

- For right now, email me hewner@cs.duke.edu – we may work out a fancier system in the future but that should work for now
- Look on the course website under “getting started” to see how to test your solution before you submit it
- Problems are due 5pm the day of class – a minimum of 2

Where We Are

- ~~1. How this course is structured~~
- ~~2. Grades~~
3. A sample problem! (warm up your computers)
 1. The solution template
 2. An actual sample problem
4. Greed! (if we have time)

The Issue of Input: An Example

The input consists of multiple test cases. Each test case is on a single line in the format: the blood type of one parent, the blood type of the other parent, and finally the blood type of the child, except that the blood type of one parent or the child will be replaced by a question mark. To improve readability, whitespace may be included anywhere on the line except inside a single blood type specification.

The last test case is followed by a line containing the letters E, N, and D separated by whitespace.

Where to Start: The Template

```
import java.util.*;

public class foo{
    Scanner in=new Scanner(System.in);
    public static void main(String[] args){
        new foo().go();
    }
    public void go(){
        int cnum=0;
        while(true){
            cnum++;
            //read input
            //do stuff
            //print output
        }
    }
}
```

Go!

- Go to the course webpage
<http://www.cs.duke.edu/courses/cps149s/fall11/>
- Do the first problem under Week 0 “Walkways”
- Work by yourself or in small groups
- Raise your hand if you’re stuck or have a question
- If you don’t have a laptop, work with somebody who does
- If you’re new, don’t worry about doing it fast – just solve the problem

Greedy Problems

- Simply a “locally good” decision that always also generates a “globally optimal solution”
- The tricky part (and this can be tricky) is figuring out what this locally good decision is, and proving to yourself it is always globally optimal



Hints

- If your dataset is large, that suggests greedy because all you need is “local information” to make a good decision so that algorithms tend to be $O(n)$ or $O(n \log n)$
- If you can think of a counterexample to your greedy solution greedy is not the way to go

For Next Week

- I hope to see everybody back!
- Do as many problems as you can from the Week 0 “Greedy” Problem list
- Do 1 problem from the Week 1 Problem list
- Bring you laptop (does anyone not have a laptop BTW?)