CIS 210: Introduction to Computer Science

Instructor: Michal Young
Graduate Assistants:
  Daniel Ellsworth
  Jason Gustafson
Winter 2012

Why come to class?

Slides will (mostly) be available after class
But ...
  Lecture is more than reading the slides, and I don’t do all the talking.

Observation: *People who skip lecture do poorly on assignments and exams*

Obtaining Course Info

Read the class web page:
  • [http://www.cs.uoregon.edu/classes/12W/cis210](http://www.cs.uoregon.edu/classes/12W/cis210)
    All basic class information is there

Follow the class blog:
    Announcements will appear there first

*Keep current! It is your responsibility.*
  *Suggestion: Subscribe to email notifications for the blog*

Textbook

*Building Java Programs*
by Stuart Reges and Marty Stepp

Read assigned chapters before lecture
come to class with questions
Experiment!
  try examples from the book, and try variations
Introduction to Computer Science

Programming is an *important part* of computer science

*Important*
It makes everything else possible.

*But just a part*
There is much more to computer science.

“*CS may be more than programming, but it is not less than programming.*”

Q: What is Programming?

A: Solving problems
The computer is a tool.
- A carpenter must know how to use a hammer, but knowing how to use a hammer doesn’t make you a carpenter.

A programming language is also a tool.
- You will learn Java. You will also learn to program.
  *Not the same thing!*

Programming is mostly about logical analysis and problem solving

Goals for CIS 210

Learn computer science concepts
Problem solving with computation

General programming skills
- includes designing programs to be understood and modified by humans
- includes testing, debugging

Expressing programs in the Java language
- but the programming concepts apply to other languages

Labs

Lab attendance is mandatory
It counts toward your grade!

Labs cover material not in lecture

It’s your best chance to understand how to solve the homework problems
Getting Help

Labs are excellent opportunities to get help
Instructor and GTFs also hold office hours. We want to
see you there!
  • But if you skip the lecture, don’t ask me to repeat it in office hours.
    I won’t do that.
Email is also useful.
  cis210-help@cs.uoregon.edu
  We try to answer quickly, usually within 24 hours.

Don’t wait to the last minute
  If the assignment is due in two days, and you are completely lost, I
  probably can’t help you much.

Other Collaboration

DO discuss the problems
  Discuss general approaches to solving them. Learn
  from each other.
If you rely on ideas from someone else, or
  somewhere else (e.g., a web site), document it in
  your solution.
DON’T copy or plagiarize
  Write every line of program code yourself.
  We can tell. We do enforce UO academic honesty
  policy.

First Assignment

On the course web site now
3 parts
  1 – “paper and pencil” (actually text file), individual
  2 – programming, individual (simple)
  3 – programming, pair exercise
Due Friday 5pm. Attach all three files to turn-in
  on Blackboard.

Pair Programming

Pair programming is allowed on some
  assignments
  • Pair programming is done with two people working
    together at one computer: One driver and one
    observer. Trade roles often.
    – Pair programming does not mean letting someone else do
      your assignment. You must understand every bit of it.
  • Keep a log of meetings.
  • Each partner turns in program listing both authors
    Always document contributions of all authors
Computers used to be bigger and more expensive ... this is part of the Bell Relay Computer

Two big ideas ...

1. Instead of connecting cables, let’s store the program in memory!
   (stored program machine)

2. Let’s write a program that simulates a more convenient computer!
   (Universal Turing Machine)
Programming with Cables

Memory Unit

Function Unit

Add content of memory cell 1532 and 2648, putting result in 2648

Programming with Cables

Memory Unit

Function Unit

Fetch content of memory cell 1532 and 2648

Programming with Cables

Memory Unit

Function Unit

Add

Programming with Cables

Memory Unit

Function Unit

Store result in memory cell 2648
Programming with Code

\[ x = x + y \]

Same thing, if memory cell 2648 is called ‘x’ and memory cell 1532 is called ‘y’:

Fetch the value in x
Fetch the value in y
Add them
Store the result in x

Universal Computing Devices

A computer executes instructions

A program is a collection of instructions

A program can simulate another (more convenient) computer.

Almost any computer can simulate any other (with the right programs).

Virtual (imaginary) Computers

Now it’s simulating a (virtual) Java computer, which executes Java programs, or a Python computer, or ...

The computer executes programs expressed in binary machine language

Introduction to Java Programming

A program is a sequence of instructions

The instructions must be written in a "programming language" like Java

Simpler, stricter, more limited than natural language
A Java program

```java
public class Hello {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
        System.out.println();
        System.out.println("This program produces");
        System.out.println("four lines of output");
    }
}
```

Its output:
Hello, world!
This program produces
four lines of output

Running a Program

Two steps: Compile, Execute

1. `examples > javac Hello.java`
2. `examples > java Hello`

Hello World!
This program produces
four lines of output

What will really happen

`examples > javac Hello.java`
Hello.java:6: `<identifier>` expected
    public static void main( String[ ] args) {
    ^
Hello.java:6: invalid method declaration; return type required
    public static void main( String[ ] args) {
    ^
2 errors

Diagnose; repair; repeat

Programming languages have rigid rules
• Expect errors. Lots.
Finally it compiles! Hoorah!
Then it crashes
You will spend a lot of time debugging.
Everybody does.
Learn to do it well. It’s part of your craft.
Anatomy of a Java program
(from Assignment 1, part 2)

/**
 * Calculates target heart rate using a standard formula.
 * CIS 210, University of Oregon, Asnmt 1 Winter 2012.
 * @author YOUR NAME HERE
 */
class THR {
    ... (more stuff here)
}

About names (identifiers)
The names of classes, methods, and variables
    Start with a letter, followed by more letters, digits, or _
Conventions in Java:
    ClassNamesAreCapitalized (camel case)
    methodsAndVariablesStartSmall
    CONSTANTS_SHOUT

Programming and CS
Why the CS major starts with programming
Learning to program is just part of CS
But programmability (universality) is the essence
You must understand programming to understand CS
Java is (just) a reasonable example to start with