

CS100J Spring 2006

- CS100J: 11 weeks of programming using Java and 2 weeks using Matlab.
 - David Gries is teaching CS100J.
 - Graeme Bailey is teaching a special section of CS100J.
- CS100M: 7 weeks of Matlab and 6 weeks of Java.

None of these versions of CS100 require previous programming experience.

Quote for the day:

"I think there is a world market for maybe five computers."
--Thomas Watson, chairman of IBM, 1943

CS100J Fall 2005

Gries:

Course outcomes:

- (1) Basic understanding of object-oriented and procedural aspects of programming, as expressed in Java.
- (2) Fluency in Java —able to write programs using classes and subclasses, as well as assignments, conditionals, and loops.
- (3) Knowledge of the basic API classes and their specifications.

CS100J Fall 2005

Methods to increase chances of success in the course.

1. Each week, the "recitation-section" will be in the ACCEL lab, where you will do guided exercises on the computer, with a TA and some consultants walking around, helping.
2. Quizzes will be given frequently. They are designed to let you know what material is important for you to know at that point. You will know quite clearly what the quiz will cover, and everyone is expected to get A on each quiz.
3. Lectures are not 45 minutes of Gries talking. You will see demos of programming and execution of programs in class almost every lecture. There will be some interactive work with you. We will try to make it interesting.
4. The course text provides an alternative way for you to learn the material; the CD at the back of the book has 250 2-4 minute lectures, each on one specific point.

Reading for this and the next lecture:

Sections 1.1, 1.2, 1.3. Lab 01 will give you practice with concepts and details of 1.2, 1.3

PLive: Lesson 0, Lesson page 1.3, Activity 1-4.1.

Summary of lectures: On course home page, click on "Handouts" and then "Outline of lectures held so far".

Topics of all lectures (this may change): On course home page, click on "Syllabus".

Today:

- Introduce expressions in Java (using DrJava)
- Show you around the CD **ProgramLive**

Programming Languages

- **Computer program:** set of instructions for a computer to perform, or execute. Written in a programming language.
- **Machine (computer) instructions:**
 - Load memory location 60000 into register 2
 - Load memory location 80310 into register 3
 - Add register 2, register 3, put result in register 2
 - If register 2 > 0, take next instruction from location 40000
- **Use high-level language:** Java, C, C++, Matlab, Algol 60.
- **Compiler:** translates program written in Java into a machine language.

Type: A set of values together with operations on them.

Memorize this definition!

Write it down several times.

Type **integer**:

values: ..., -3, -2, -1, 0, 1, 2, 3, 4, 5, ...

operations: +, -, *, /, unary -

Type: A set of values together with operations on them.

Memorize this definition!
Write it down several times.

Type **integer**:

values: ..., -3, -2, -1, 0, 1, 2, 3, 4, 5, ...

operations: +, -, *, /, unary -

-2^{31} . It uses 32 bits

Type **int**:

values: -2147483648, -2147483647, ..., -3, -2, -1, 0, 1, 2, 3, 4, 5, ..., 2147483646, 2147483647

operations: +, -, *, /, unary -

Type: A set of values together with operations on them.

Type **double**:

mantissa

exponent

values: Examples: $-22.51E6$ equivalent to -22510000
or $-22.51 * 10^6$
 $22.51E-6$ equivalent to $.00002251$
or $22.51 * 10^{-6}$

An approximation to the real numbers.

operations: +, -, *, /, unary -

Precedence of operators (page 23)

- Unary operators: + - !
- Binary arithmetic: * / %
- Binary arithmetic: + -
- Arithmetic relations: < > <= >=
- Equality relations: == !=
- Logical and: &&
- Logical or: ||

The lecture also touches on:

- **Types boolean and String.**
- **Assignment statement.**

You will use these things in Lab 01.