

Syllabus - Ist Course @ De Anza

Introduction to Computer Programming Using Java

Office hours

Location - F51e - Mon/Wed - 1:00pm to 2:50pm. You can also call me at 408 864 5566 during my office hours.

Faculty Information

Sukhjit Singh phone: 408 864 5566 email: singhsukhjit@fhda.edu Office Location: F51e

Requisites: (Students may receive credit for either Computer Information Systems (36A and 36B) or 35A, but not both.)

Advisory: English Writing 211 and Reading 211 (or Language Arts 211), or English as a Second Language 272 and 273; Mathematics 114 or equivalent.

Hours: Four hours lecture, one and one-half hours laboratory (66 hours total per quarter).

Description: An introduction to computer programming. The primary objective is to teach problem solving using the Java programming language. Emphasis will be placed on structured procedural programming with an introduction to object-oriented programming. Designed primarily for computer science and related transfer majors.

Student Learning Outcome Statements (SLO)

- Design solutions for introductory level problems using appropriate design methodology incorporating elementary programming constructs.
- Create algorithms, code, document, debug, and test introductory level Java programs.
- Read, analyze and explain introductory level Java programs.

Course Objectives

- A. Illustrate the difference between procedural and object oriented programming.
- B. Describe steps in software development life-cycle design, development, styles, documentation, testing, and maintenance.
- C. Explain the Java Runtime Environment (JRE) and Java Software Development Kit (SDK).
- D. Illustrate declaring identifiers of different data types.
- E. Use data types to declare variables in Java programs.
- F. Apply input and output functions to read data using keyboard and output to screen.

- G. Use expressions, statements and operators to construct program building blocks that compute values.
- H. Apply control structures to break up flow of program execution and conditionally execute blocks of code.
- I. Illustrate usage of static methods to improve clarity, quality and development time of a computer program.
- J. Develop programs using functions that enable input and output from text files.
- K. Demonstrate usage of arrays to process variety of data problems.

Expanded Description: Content and Form

- A. Illustrate the difference between procedural and object oriented programming.
 - 1. Programming language overview.
 - 2. Compare and contrast procedural versus object oriented programming languages.
 - 3. Language generations – First to Fifth generations.
 - 4. Compiled versus interpreted languages.

- B. Describe steps in software development life-cycle design, development, styles, documentation, testing, and maintenance.
 - 1. Flowcharts
 - 2. Pseudocode
 - 3. Software Development Life Cycle
 - 4. Coding conventions
 - 5. Documentation

- C. Explain the Java Runtime Environment (JRE) and Java Software Development Kit (SDK).
 - 1. import directive with java.lang package
 - 2. Parts of a Java program
 - 3. JRE program processing

- D. Illustrate declaring identifiers of different data types.
 - 1. Constants
 - 2. Variables
 - 3. Class names

- E. Use data types to declare variables in Java programs.
 - 1. Primitive types
 - a. Integer
 - b. Floating point
 - c. Boolean
 - d. Character
 - 2. Classes

3. Declaring variables
4. Scope of variables
5. String and Date class

F. Apply input and output functions to read data using keyboard and output to screen.

1. Scanner
2. System.out.println and System.out.printf

G. Use expressions, statements and operators to construct program building blocks that compute values.

1. Arithmetic and Unary operators
2. Functions in java.lang.Math

H. Apply control structures to break up flow of program execution and conditionally execute blocks of code.

1. Relational Operators
2. Logical operators
3. The 'if' statement
4. Flags
5. Expanding the 'if' with 'else' and 'else if'
6. Comparing strings
7. The 'switch' statement
8. Loops
 - a. while loop
 - b. do while loop
 - c. for loop

I. Illustrate usage of static methods to improve clarity, quality and development time of a computer program.

1. Writing static methods definition.
2. Passing argument to parameters.
3. Pass by value.

J. Develop programs using functions that enable input and output from text files.

1. Output to a text file.
2. Input from a text file.

K. Demonstrate usage of arrays to process variety of data problems.

1. One-dimensional arrays
 - a. Accessing array elements.
 - b. Initialization
 - c. Processing array contents.

2. Sequential search
3. Selection sort

Grading System for this course

For Letter Grade:

- Grade: A+ assigned with 97% or higher
- Grade: A assigned with 93% or higher
- Grade: A- assigned with 90% or higher
- Grade: B+ assigned with 87% or higher
- Grade: B assigned with 83% or higher
- Grade: B- assigned with 80% or higher
- Grade: C+ assigned with 77% or higher
- Grade: C assigned with 73% or higher
- Grade: D+ assigned with 70% or higher
- Grade: D assigned with 63% or higher
- Grade: D- assigned with 60% or higher
- Grade: F assigned with 0% or higher

For Pass/No Pass:

- Grade: Credit assigned with 70% or higher
- Grade: No Credit assigned with 0% or higher
- Incomplete

Audit

Withdrawal

Grading

- Labs - 50% of the grade
- Midterm - 25% of the grade
- Final - 25% of the grade

Assignments Due Dates:

You will be assigned 8 to 10 assignments during the quarter. Assignment details can be found on the Assignment page. Midterm and Final dates can also be found on Assignment page

Methods of Evaluating Objectives

- A. Evaluation of programming assignments for correctness, use of design principles, documentation, efficiency and teamwork.
- B. One or more examinations requiring some programming, concepts clarification and exhibiting mastery of programming principles.
- C. A final examination requiring concepts clarification and exhibiting mastery of programming principles.

Texts and Supporting References

Introduction to Java Programming, Comprehensive (10th Edition) [Paperback] - Y.

Daniel Liang (Author)

ISBN-13: 978-0133761313 ISBN-10: 0133761312

Link to [Amazon.com](https://www.amazon.com)

Earlier editions of the same text are fine to use. As long as it is 6th edition or later.

Another great reference is [Thinking in Java](#) by Bruce Eckel.

Attendance

You are responsible for completing all work assigned in this class in a timely fashion. You do not have to contact me with a reason of absence.

You should be enrolled in the class at De Anza College for getting course access and to attend the class.

Withdrawing

Once you are added to the class it is your responsibility to withdraw. I will not drop you from the class. The earned grade will be assigned at the end of the quarter.

Academic Dishonesty

You are encouraged to discuss the ideas presented in the class. Copying or Cheating of work will result in zero grade for that assignment and may result in a failing grade. Basically I cannot tolerate cheating. You must work your solutions independently and all assignments and tests should be your own original work NO MAKEUP TESTS WILL BE GIVEN. You must pass the final to get a passing grade in this class.

Submitting Lab Assignments

All assignments must be submitted electronically using the following guidelines.

Pl. email your assignments to **cislabs05@gmail.com**

Include the following information in the subject line

1. Your section #
2. Lab #
3. Your legal name (as it shows on academic records)

Use Text files for everything you submit.

You may submit files only with the following extension

- .txt (any design notes you want me to look at)
- .java (your source code.)
- .jpg or .gif (if you use any images for graphics programming)

You must include a readme.txt (for lab4 onwards) providing instructions to review and run your code.

Adequately test your code and run the test run of your code in a file called

testrun.txt.

Every file should have the following information

Your Name

Class and Section

Assignment Number

Due Date

Date Submitted

If you submit more than one file you must use winzip to compress all files into a single zip file and submit.

All Assignments are submitted by email to cislabs05@gmail.com.

Subject with each submission should be stated as - "CIS 36a - Lab <#>" - Replace # with the assignment number you are submitting.

Lab Grading Criteria

Full programming assignments will be evaluated with consideration given to

Accuracy (does the program solve the computing problem)

Adherence to Object Oriented Programming Methodology techniques (for Assignment 2 onwards)

Code readability and appearance

Naming Conventions

Documentation

Timeline

Professional Presentation

Software

Download Java Standard Edition (latest version). Follow the installation instructions provided on the same page.

Mac users have java pre-installed and available in the Unix Shell on Mac OS. If you prefer a GUI based IDE then work with Eclipse.

Here is a video that might help - <http://www.youtube.com/watch?v=Otlva4ZHfqc>

Lab Topics

A. Debug code and output results of execution.

B. Write and/or debug code with input from keyboard and output to monitor.

C. Write and/or debug code implementing arithmetic expressions.

D. Write and/or debug code employing decision concepts and selections statements.

E. Write and/or debug code employing repetition concepts and for, while and do while loop structures.

F. Write and/or debug code implementing functions.

G. Write and/or debug code implementing file I/O.

H. Write and/or debug code implementing one-dimensional arrays.

