Syllabus

Objective

The aim of CS111 is to introduce the student to the fundamental techniques used in computer science and software development. Upon completion of this course, the student should be able to design, write, test, and analyze programs to solve simple real-world problems. CS111 uses the Java programming language, but it is not a training course in Java. Many Java features, such as applets, GUI programming, Java Server Pages (JSP), etc., are not covered in this course.

Prerequisite Knowledge

Parts of a computer: processor, memory, disk drives, screen (monitor), keyboard, mouse File system concepts: files, directories (folders), difference between data files and programs Internet skills: use of web browser and e-mail Algebra concepts: variables, functions, exponentiation, logarithms

Concepts

- The programming process

- Algorithm design

- Input, output, and temporary storage (variables)

- Mathematical computation

- Boolean logic

- Decision-making (branching)

- Iteration (looping)

- Test case construction

- Object Oriented Programming

- Analysis of Algorithms

Introductory Java Programming

- The compilation process

- Compile-time and run-time errors

- Objects and references

- Java Basic program structure and syntax rules

- Data types: boolean, int, double, char, objects/references

- Literals, variables, constants, scope of variables

- Operators, expressions, operator precedence

- Type conversion

- Instantiating objects, primitive and reference types

- null keyword

- Calling methods (static and non-static)

- Decision statements: if, if/else, if/else if/.../else

- Loops: do/while, while, for

Procedural Programming

- Procedures (a.k.a. subroutines, functions, methods, etc.)

- Solving problems using recursive procedures

- Method frames, caller/callee, the call stack

Object Oriented Programming

- Object-oriented design, encapsulation

- Defining classes, member variables, and methods

- Public and private modifiers

- Garbage collector

Data Structures

- Arrays

Efficiency of Algorithms

- Efficiency analysis, big-O notation

- Sequential search, binary search

- Sorting: selection sort, insertion sort, mergesort using arrays