

Course Topics
EC327 – Introduction to Software Engineering
Fall 2020

The following is a tentative topics list for EC327. The actual material covered will be depend on class intangibles, such as average student background and progress, and may follow a different order.

Introduction (1-2 lectures)

- **Overview**
 - The six-level computer
- **Program development cycle**
 - Design, pre-processing, compilation, linking, loading
- **Operating Systems**
 - Linux (Unix)
- **Language levels**
 - Simplified assembly
 - Bit/byte arithmetic and logic
 - Registers and instructions

Fundamentals (8-9 lectures total)

- **Introduction (2 lectures)**
 - Data primitives, constants
 - Selection (if, switch)
 - Loops (for, while, nested)
 - Basic string type
- **Functions (2-3 lectures)**
 - The call stack
 - Scope
 - Pass by value vs. by reference
 - Overloading
 - Prototyping and header files
 - Inline functions and macros
- **Arrays (< 1 lecture)**
- **Pointers (2 lectures)**
 - Pointers and arrays
 - Dynamic memory allocation
 - Heap, stack memory allocation
- **File I/O (< 1 lecture)**

Object-oriented programming (9-10 lectures total)

- **Classes**
- **Objects**
 - Dot operator
 - Scope
 - Constructors/destructors
 - Dynamic creation
 - Copy constructors (deep/shallow)
- **Overloading, overriding operators**

- *this* pointer
- The string class
- Inheritance
- Polymorphism, virtual functions, dynamic/static typing

More OOP (2-3 lectures)

- Exception hierarchy
- Vectors and templates
- Standard Template Library (STL)

Program development (*throughout the homework assignments and labs*)

- Makefiles
- Programming style
 - Variable names
 - Commenting
- Integrated Development Environments
 - Version control (e.g., GitHub, Google Code)
 - Libraries
 - Examples of IDEs
- Debugging (1 dedicated lecture)
 - gdb
 - Techniques
- Optimization
 - Basic coding tips
 - Compiler flags

Data structures and algorithms intro (1-2 lectures)

- Big O notation (basics)
- Data structure examples
 - Linked lists
 - Stacks
 - Queues
- Sorting

More topics (as time permits)

- C vs. C++
- More on Templates and STL
- Threads
- Android/Mobile platform/Web programming (*through the course project*)
- Function pointers
- Multiple inheritance
- Design Patterns
 - Model-View-Controller, Singleton
- Hacking
 - Stack/heap/format overflows
- C++ variants and sister languages
 - C#, Java, Objective-C, SystemC