EEC120 Teaching/Evaluation Plan

Week	Lecture	Laboratory	Notes
Week 01	• Introduction to EEC120 Computer Programming I • Introduction to Computer Systems 1. Hardware (CPU, input, output) 2. Software (Kernel, OS, userspace) 3. Numbers & Logic • Introduction to Computer Programming 1. Algorithms: instruction sequencing 2. Flow-charts: visual representation 3. Interpreters ⇒ scripts 4. Compiler and Linkers ⇒ binary executables	• Tools (compiler/IDE) Installation/Setup	• **[VIDEO GUIDE]**???
Week 02	• Introduction to C language 1. Basic code structure (syntax) 2. Function and code-block 3. Simple text display • Creating first executable binary 1. Syntax (language) and semantic (decision logic) errors 2. Simple debugging	• Explore building binary from c source code • Explore creating text decorations using symbols	• **[VIDEO GUIDE]**???

Week	Lecture	Laboratory	Notes
Week 03	• C Variables (data storage) 1. data types and sizes • C Operators 1. arithmetic and logic 2. boolean vs bitwise • Data processing 1. equations and tasks 2. (optional) math.h library functions	• Understanding storage types and practical use • Exploring code that does equations/calculations	**Assignment 1 Queue**
Week 04	• Flow control: Branches (selection) 1. `if-else` structure 2. `switch-case` structure 3. (otional) the `?:` combo	• Understanding decision making process in code • Exploring code that control program flow	
Week 05	• Flow control: Loops (repetition) 1. `for` structure 2. `while` structure 3. `do-while` structure	• Understanding iterative flow in code • Exploring code that repeats itself	**Assignment 1 Due (5%)**
Week 06	• C (Data) Array 1. storage in groups 2. accessing array items (content)	• Understanding array as group-storage • Exploring code that process on arrays (read/write)	**Lab Assessment 1 (10%)**
Week 07	• C String 1. array of chars (null terminated) 2. (optional) string.h library functions	• Understanding string as char array • Exploring code that process on strings (read/write)	**Mid-term Examination (20%)**

Week	Lecture	Laboratory	Notes
Week 08	• C Pointers 1. indirect data access (data that holds address to other data) 2. array name as address 3. pointers in loops 4. pointer arithmetic	• Understanding pointers as method for indirect access • Exploring code that deals with (or use) pointers	**Assignment 2 Queue**
Week 09	• C Functions 1. function return value 2. function parameters 3. (extra) data/variable scope and lifetime	• Understanding functions as building block for modular programming • Exploring code that has multiple functions	
Week 10	• File access (permanent storage) 1. accessing file on secondary memory (i.e. disks) 2. text vs binary file	• Understanding pros/cons of secondary storage • Exploring code that read/write from/to files	**Assignment 2 Due (5%)**
Week 11	• Practical Applications ◦ requirements and considerations • C struct 1. data group 2. accessing data in struct	• Lab Project • (extra) use struct to create abstract data type • (extra) creating functions to process struct (towards oop?)	**Lab Assessment 2 (10%)**
Week 12	• Practical Applications (cont.) ◦ requirements and considerations	• Lab Project (cont.)	
Week 13	• Practical Applications (cont.) ◦ requirements and considerations	• Lab Project (cont.)	

Week	Lecture	Laboratory	Notes
Week 14	• Practical Applications (cont.) ◦ requirements and considerations	• Lab Project (cont.)	**Lab Project Due (10%)**

From:
https://azman.my1matrix.cc/ - **Azman @MY1**

Permanent link:
https://azman.my1matrix.cc/doku.php?id=archive:eec120plan

Last update: **2026/08/11 09:16**