**Week 1:**

Topics: introduction, C fundamentals, input/output, Unix Commands, compile and run C programs, variables, macro definitions, identifiers, formatted input/output

Reading: Textbook Chapter 1, 2, and 3, Linux Command Line book Chapter 1 and 2

Homework exercises: (Chapter 2) Exercises #5, (Chapter 3) Exercises #4 and #i5

**Week 2:**

Topics: Unix commands, review selection and loop statements, Lvalue, increment and decrement operators in complex statements, boolean values (0 and 1) in C, break, continue, and null statements,

Reading: Textbook Chapter 4, 5, and 6, Linux Command Line book Chapter 3 and 4

Homework exercises: (Chapter 4) Exercises #9 and #12, (Chapter 5) Exercises #1a,b, #2, and #6 (Chapter 6)  Exercise #11, 12, 13, and 14

**Week 3:**

Topics: Unix commands, integer and floating point types, type conversion, character type, reading and displaying characters

Reading: Textbook Chapter 7, Linux Command Line book Chapter 12 (vi)

Homework exercises: (Chapter 7) Exercises #11, #12, (Chapter 7) Programming Projects #11 and #12

**Week 4:**

Topics: Unix commands, arrays, functions, function declaration, arguments, recursive functions.

Reading: Textbook Chapter 8 and Chapter 9

Homework exercises: (Chapter 8) Programming Projects #13 and #15, (Chapter 9) Exercises #7, 8, 9, 11, 15, 18, and 19, Programming Projects #4 (project 16 of chapter 8 is in-class exercise).

**Week 5:**

Topics: Local vs external variables, block scope, organizing a C program, pointer variables and assignment, pointers as arguments and return values.

Reading: Textbook Chapter 10 and Chapter 11

Homework exercises: (Chapter 10) Exercises #1 and #2, Programming Projects #2, (Chapter 11) Exercises #2, 3, 5, Programming Projects #4

**Week 6:**

Topics: pointer arithmetic, using Pointers for Array Processing, combining the **\*** and **++** operators, using array name as a pointer, pointer as array arguments, segmentation fault

Reading: Textbook Chapter 12

Homework exercises: (Chapter 12) Exercises #1, 3, 5, 8, 10 Programming Projects #1, 2, 3, 4, 5

**Week 7:**   
Topics: String literals, string variables, string library functions, array of strings and command-line arguments

Reading: Textbook Chapter 13

Homework exercises: (Chapter 13) Exercises 4, 12, 15, 18, programming projects 4, 9, 10, 14(use int instead of bool for return type)

**Week 10:**

Topics: Read/write text data to/from files, obtaining file names from the command line, formatted file input and output

Reading: Textbook Chapter 22

Homework exercises: (Chapter 22) Exercises 13, 14, programming projects 2, 4, 17, 18

**Week 11:**

Topics: Access the members of a struct, write functions that take structs as parameters, write functions that return structs, nested arrays and structures, program design.

Reading: Textbook Chapter 16

Homework exercises: (Chapter 16) Exercises 3, 5, 7, programming projects 1, 5

**Week 12:**

Topics: Memory allocation functions, dynamically allocated arrays, strings, and structures, deallocating storage, linked list

Reading: Textbook Chapter 17

Homework exercises: (Chapter 17) Exercises 2, 3, 4, 7, 8, 12, programming projects 3, 4

**Week 13 & 14:**

Topics: Type definition, header files, conditional compilation for protecting header files, diving a program into multiple files, building a multiple-file program, Makefiles

Ordered linked list, program design, stack, function pointers as argument, using the qsort function in the standard C library.

 Reading: Textbook Chapter 15 and Chapter 7 (7.5 Type Definitions), Chapter 17 and 19

Homework exercises: (Chapter 15) Exercises 2, 5, 6, programming projects 4

(Chapter 17) Exercises 13, 15, 17, 18, (Chapter 19) Exercise 1 programming project (Chapter 19) 2