C programming **Spring 2018**

The review for the final examination includes the following topics per chapters:

Chapter 1: Variables, Arithmetic Expressions; "for" statement; Symbolic constants; Character Input and Output; File Copying; Character, line, and word counting; Introduction to Arrays, Functions, and Pointers; Arguments - Call by value; External Variables and Scope.

Chapter 2: Types, Operators and Expressions: Names; Data Types and Sizes; Constants; Declarations; Arithmetic Operators; Relational and Logical Operators; Conversions; Increment Decrement Operators; Bitwise Operator; Assignment Operators and Expression; Conditional Expressions; Precedence and Order of Evaluation

Chapter 3: Control Flow; Statements and Blocks; If-Else; Else-If; Loops - While and For Loops - Do-While; Break and Continue; Goto and labels.

Chapter 4: Functions and Program Structure; Basics of Functions; Functions Returning Non-integers; External Variables; Rules; Header Files; Variables; Block Structure; Use of Macros and Conditional Inclusion.

Chapter 5: Pointers and Arrays; Pointers and Functions; Pointers to Pointers; Multi-dimensional Arrays; Pointers to Functions.

Chapter 6: Structures; Structures and Functions; Arrays of Structures; Pointers to Structures; Typedef; Unions; Bit-fields.

Chapter 7: Input and Output; Formatted Output – printf; Argument Lists; and File Access.

Practical problems to solve in C programming:

26. In a C Programming exam, each exam is graded by two GTAs. If the difference of their grades is less than z, the final grade is their average. Otherwise, the test is reviewed by a third GTA such as: i) If the grade of the third GTA is equal to the average of the first two grades, that is the final grade. ii) If the grade is less than the minimum of the first two grades, the final grade is minimal. iii) Otherwise, the final grade is the average of the grade of the third reviewer and the one of the first two grades closest to it. Write a C program to read the two grades and the difference z and to display the final grade according to that method." If the grade is less than the minimum of the first two grades, do you want the program to display the word "minimal" or do you want it to display the lowest grade?

27. Given integer values of x and y, you need to generate 20 random values within the interval [x, y]. Print the array. Determine the index of the largest even negative number in it (ranging from 0 to 19). Account for the possibility that there might be no such number in the array. Calculate the sum of all positive odd numbers. You are only allowed to go through the array values once. You should define separate functions for generating random numbers and checking even and odd numbers.

28 Given the values of three censors that check the condition of machine on a scale of 0 to 40: Sum these values. Calculate the total percentage, i.e. Sum / (40 * 3)) * 100. Depending on the percentage: i) If the percentage is less than 24, make sure to report that the machine is malfunctioning; ii) If the percentage is

less than 7, make sure to report the machine is running idle; and iii) If the percentage is greater than 42, make sure to report that the machine is operating normally. Provide test outputs for all possible reports.

29. Write a C program that uses two pointers to read two float numbers first and then to swap the values they point to. Then, use the same pointers to output both the largest value and the smallest value.

30. Write a C program that accepts a set of 15 digits and then accepts a number and tells you the position of the number in the set of numbers or says that the number is not in the set if it is not in the set.

31. Write a C program to assign random integers to a 2×6 array and displays the maximum and the minimum value of each row and column. Use pointer arithmetic to process the array.

32. Write a C program to use a pointer variable to read a double number and to display its fractional part. For example, if the user enters -56.47, the C program is to display 0.47.

33. Create a void function to take as parameters an array which contains the prices of some shoes in a shoe store and their number and use proper variables to return the highest, the lowest, and the average of the prices. Write a C program to read the prices of less than 60 shoes and stores them in an array. If the user enters -1, the insertion of prices is to be terminated. The C program needs to use a C function to output the highest, the lowest, and the average of the prices.

34. Write a C program to compute and to print explicitly all the roots of the following quadratic equation $\mathbf{x^2 - 2x - 4 = 0.}$

35. Write a C program to use three pointer to read three integers and to check if they are in ascending order such as -20, -19, -18. The program is an user-defined program that asks the user to enter negative numbers.

36. Use **ptr1, ptr2**, and `temp` and complete the program below with the purpose to reverse the elements of **arr**. Then, use **ptr1** to display the array elements. For example, the new **arr** is to be: **2.5 9.4 -3.8 -4.1 1.3:**

```
   #include <stdio.h>
   int main(void) {
   double arr[] = {1.3, -4.1, -3.8, 9.4, 2.5}, temp, *ptr1 = arr, *ptr2 =
arr+4;
   ...  }
```

37. Create a C function to compute the largest common divisor of two positive whole numbers. Assume two integers p and q, where p > q. If q divides p precisely, this is the largest common divisor. If the remainder z of the division p/q and it is not 0, then we divide q with z. If the new remainder of the division is 0, then the LCD is z; otherwise, this process is repeated. Write a C program to read two integers and to use the C function to compute their largest common divisor.

38. Write a C program to read a string of less than 16 characters, and if it ends with **ee**, the program is to display it in reverse order.

39. Write a C program to read characters and to store them in an array of 12 places with the restriction that no duplicated character is to be stored. If the user enters `'n'` or the array is full, the insertion of characters is to terminate.

40. Use C function to take an integer parameter and returns the result of **`13 + 23 + 33 + ... + n3.`** Write a C program to read a positive integer up to 17 and to use the function to display the result of the expression.

41. Write a **void** function that takes as parameters the coefficients of a quadratic trinomial and returns its real roots, if any. Write a program that reads the coefficients of a trinomial (e.g., a, b, and c) and uses the function to solve the equation. The program should force the user to enter a nonzero value for a. It is reminded from algebra that in order to find the roots of the trinomial $ax^2 + bx + c$ with $a \neq 0$, the discriminant $D = b^2 - 4ac$ is tested.

a. If $D > 0$, it has two real roots $r_{1,2} = (-b \pm \sqrt{DD})/2a$.
b. If $D = 0$, it has one real double root $r = -b/2a$.
c. If $D < 0$, it has no real root.

42. Write a C function to take as parameters two strings and to return a pointer to the longer string. If the strings have the same number of characters, it should return NULL. Write a C program that reads two strings of less than 11 characters and uses the C function to display the longer one.

43. Write a C program to display the MAC (Medium Access Control) address in lines, where each line is to contain 16 bytes in **`hex`** hexadecimal format. The MAC address consists of six octets (bytes) and it is to be entered in the following **`x.x.x.x.x.x`** format so that each x is an integer in the interval [0, 255].

44. Write a recursive function that uses the **`strchr()`** function to display the number of occurrences of a character inside a string. Create a C program to read a string of less than 10 characters and a character and calls the function.

45. Write a C function to take as parameters three 2 x 4 two-dimensional arrays, computes the sum of the first two, and stores that sum into the third one. Create a C program to read 16 integers, to store them in two 2 x 4 two-dimensional arrays, and uses the function to display their sum.

46. Create a C function to accept a variable number of pointers to integer arguments and to return the pointer to the largest number. Create a C program to read three integers and to use the function to display the largest value.

47. Create a C function to take as parameter an array of integers and uses the insertion sort algorithm to sort it in ascending order. Write a C program to read five integers, to store them in an array, and to use the function to sort the array.

48. Create a C program to read the names of 7 students (less than 10 characters each), stores them in a two-dimensional array, and uses the bubble sort algorithm to sort the array in alphabetical order. The program is to display the array before it ends.

49. Define the structure type **`circle`** with member the radius and the type square with member the length. Write a C function to take as parameters a pointer of type **`void*`** and an integer parameter. If its value is 0, the pointer points to a structure of type **`circle`**, otherwise to a structure of type **`square`**. The function should increase the member of the pointed structure by 5 and return that pointer. Write a program to test the function.

50. Create a C program to declare an array of 5 pointers to strings and to allocate the exact memory needed to store strings of less than 8 characters. The program should read 5 strings and display the longest one. If more than one string is the longest, the program is to display the one found first.