Problem 1:

Write the “Hello world” program that you have on the theory slides, compile and run it to familiarize yourself with the C work environment.

Solution: hola-mundo.c

Problem 2:

Modify the previous program to read a name from the keyboard and print the name on the screen. For example, if said name is Juan, your program will display “Hello Juan”.

Solution: hola-nombre.c

Problem 3:

Create a C program that declares two integers and a float, assigns value to them, and prints them to the screen. Define is used.

Solution: print-simple.c

Problem 4:

Create a C program that declares an integer, a long int, and a float, assigns them a value, and prints them to the screen in different formats. Then read the two integers separated on the entry line by a ",".

Solution: scanf-printf.c

Problem 5:

Create a program that takes a name and an age as arguments and prints them to the screen. If the arguments are not there you should print a command format message and exit.

Solution: persona.c

Problem 6:

Create a program that takes an integer as an input argument, declares a 10-element vector, initializes it with the argument, and adds the 10 elements of the vector using first for and then while. The results of both sums will be printed on the screen.

Solution: sumadigitosarray.c

Problem 7:

Create a program that receives two real numbers as input and prints their sum to 3 decimal places. To add them, a sum function must be declared that adds both numbers and returns the result.

Solution: sumarf.c

Problem 8:

Create a program that prompts for multiple strings, each of which can have blanks, and counts the number of blanks and the total number of characters in all strings.

Solution: arrayCadenas.c

Problem 9:

Create a program that declares an array with 5 integer elements and that, using for loops, prompts for their values on the keyboard and then prints the array to the screen.

Solution: array.c

Problem 10:

Create a program that given 2 numbers (dividend and divisor) calculates their integer quotient, their remainder, their modulus, their square root, their real division, and their upper and lower rounding, and prints them on the screen. You must use the "math" library.

Solution: operacionesMat.c

Problem 11:

Create a program that declares two integer pointers and a string. This program should print the addresses of the pointers and then use malloc and free to allocate space to those pointers and assign them a value.

Solution: pointers.c

Problem 12:

Write a C program that swaps the values of two numbers using reference arguments in an swap function.

Solution: intercambiar.c

Problem 13:

Write a C program that allows you to print one string, fall asleep for n seconds (passed as an argument), and then print another.

Solution: printf-delayed.c

Problem 14:

Write a program to use sizeof and strlen. Write a program that displays the size in bytes of a 10-element integer vector. And declare a string of 12, copy a text of 8 and show its length.

Solution: sizes.c

Problem 15:

Write a C program that allows you to create a dynamic array of integer type whose size is requested by input, as well as its elements. Once created, the array should be printed in reverse order.

Solution: arraydinamico.c

Problem 16:

Write a program in C that for a number of students, which is requested by keyboard, asks for each one of them: name and 3 notes (which can have decimal places). Once the list of students has been entered, it calculates the average of the three grades for each student and displays the name and average grade of the student on the screen.

Solution: notaalumnos.c

Problem 17:

Make a C program that implements an integer calculator. Two integers must be read per keyboard. A menu must be printed on the screen that shows the following options: addition, subtraction, multiplication and division, and finally the Exit option. The calculator will run indefinitely until the Exit option is chosen. The following functions must be performed:

- int menu (void);

Displays a menu on the screen with the options: 1. Addition of broken, 2. Subtraction of broken, 3. Multiplication of broken, 4. Division of broken, 5. Exit. The user will be prompted to choose one of the options and return it to the main program.

- void Ask_numbers (int * n1, int * n2);

It will prompt the user for the two integers and return that reading by reference.

- int add (int n1, int n2);

It receives two integers as an argument, performs the sum of them and returns the result.

- int subtract (int n1, int n2);

- It receives two integers as an argument, subtracts them and returns the result.

- int multiply (int n1, int n2);

It receives two integers as an argument, multiplies them and returns the result.

- int divide (int n1, int n2, int * coc, int * remainder);

It receives two integers as an argument, performs the division of them and returns the quotient and remainder by reference.

Create a calculadora.h file with the headings of the functions and a calculadora.c file with the code of the functions without including the main program. The main program must be carried out in the file mainCalculadora.c

Solution: calculadora.zip

Problem 18:

Write a C program that reads create a 4-row x 3-column integer matrix, prompts for the values on the screen, and prints it.

Solution: matriz.c

Problem 19:

Write a C program that displays the parameters that a C process receives on the screen.

Solution: parametrosmain.c

Problem 19:

Write a C program that declares and integer pointer, allocate to the pointer a memory space of 10 ints and assign values to the vector created. To se if everythong is find, print the vector in the STDOUT.

Solution: malloc.c