**PROGRAMACION APLICADA A LA AUTOMATIZACION INDUSTRIAL**

**PRACTICA # 3 Desarrollo de aplicaciones en C que manejen colecciones de datos.**

**CAPÍTULO DEL CURSO:** PROGRAMACIÓN ESTRUCTURADA PARA APLICACIONES DE AUTOMATIZACIÓN.

**TEMA DE LA ACTIVIDAD:** USO DEL LENGUAJE C EN APLICACIONES SIMPLES CON FUNCIONES, PUNTEROS, Y ARREGLOS.

**NOTA**

**OBJETIVOS DE APRENDIZAJE:**

* Elaborar algoritmos en lenguaje C que permitan resolver los problemas prácticos empleando funciones, punteros, y arreglos.
* Usar el ambiente de desarrollo integrado de Qt Creator para la edición, compilación, ejecución y depuración de programas en C.

**DURACIÓN:** 120 minutos

**MATERIALES Y HERRAMIENTAS:**

* Computadora con Qt Creator correctamente instalado

**INTRODUCCIÓN**

El lenguaje de programación C, es uno de los pocos lenguajes de alto nivel que permite el manejo de las direcciones de memoria hasta cierto punto, lo que permite hacer programas más eficientes. También presenta, al igual que la mayoría de los lenguajes, el uso de arreglos de memoria para trabajar más cómodamente con colecciones de elementos de un mismo tipo, y presenta estructuras para manejar colecciones de elementos de diferentes tipos. En esta parte, se realizan ejercicios que emplean punteros, arreglos y estructuras. Se debe emplear como bibliografía la conferencia III o sesión de clase 3 y la propia bibliografía allí propuesta.

**DESCRIPCIÓN DE LA PRÁCTICA:** El estudiante prueba los programas realizados en la prepráctica en el ambiente de desarrollo Qt Creator. Luego modifica o agrega funciones a algunos de estos programas con control de flujo realizados previamente para cumplir un nuevo objetivo. Posteriormente, estos programas son implementados y depurados en el ambiente de desarrollo.

**PROCEDIMIENTO:**

1. Crear una carpeta en el escritorio que tenga de nombre “practica\_2” donde guardará sus programas y luego podrá llevárselos con usted. (Si se hace en su PC propia puede obviar este paso, pero debe saber dónde se guardarán sus proyectos.)
2. Abra el IDE o Ambiente de Desarrollo Integrado que utilizará (Qt Creator).
3. Implemente y pruebe la solución de la pregunta #1 editando su programa en el IDE. Resuelva cada pregunta en un proyecto independiente. Compile y ejecute su programa en el IDE. En caso de no compilar o ejecutarse adecuadamente revise los mensajes de error (en caso de error de sintaxis) o depure el programa (en caso de error de ejecución). Copie el código del programa al informe de esta práctica (como texto) y la ventana de consola dónde se vea su ejecución de forma válida (como imagen).
4. Implemente y pruebe la solución de la pregunta #2 editando su programa en el IDE. Compile y ejecute su programa en el IDE. Copie el código del programa al informe de esta práctica y la ventana de consola dónde se vea su ejecución de forma válida.
5. Implemente y pruebe la solución de la pregunta #3 editando su programa en el IDE. Compile y ejecute su programa en el IDE. Copie el código del programa al informe de esta práctica y la ventana de consola dónde se vea su ejecución de forma válida.
6. Informe al profesor cuando haya concluido esta etapa.
7. Realice las modificaciones que se encuentran en las preguntas #1 y #2 posteriormente en este documento.
8. Vuelva a realizar los pasos 3, 4 y 5 con los programas ya modificados.
9. Informe al profesor cuando haya concluido esta etapa.
10. Describa las conclusiones y recomendaciones de la práctica.

Suba el informe a Sidweb.

**EJERCICIOS:**

1. **Realice un programa que declare las variables de punto flotante numero1 y numero2, con numero1 inicializado a 7.3, definir la variable alphanum de tipo caracter. Luego realice las siguientes acciones:**
* Declare la variable fPtr que sea un puntero a void.
* Solicite por teclado el ingreso de valor para la variable alphanum
* Apunte con fPtr a la variable numero1 (asigne a fPtr la dirección de la variable numero1).
* Imprima en pantalla el valor de la variable apuntada por fPtr.
* Asigne el valor de la variable apuntada por fPtr a la variable numero2.
* Imprima en pantalla el valor de la variable numero2.
* Imprima en pantalla la dirección de numero1.
* Imprima en pantalla la dirección almacenada en fPtr.
* Imprima en pantalla la dirección de numero2.
* Apunte con fPtr a la variable alphanume (asigne a fptr la dirección de la variable alphanume)
* Imprima en pantalla la nueva dirección almacenada en fPtr.
* Imprima en pantalla la dirección de alphanume.
* Imprima en pantalla el valor de la variable apuntada por fPtr.

**¿Por qué se declaró a fPtr como un puntero a void? ¿Qué tipo de casteo es necesario para imprimir los valores apuntados por el puntero fPtr? Explique detalladamente su respuesta.**

1. **Realice un programa que permita cambiar los valores de las diferentes posiciones de un arreglo considerando lo siguiente:**
* Defina las variables rangoini y rangofin de tipo entero.
* Defina un arreglo de nombre array que permita amanerar 10 números de tipo entero.
* Solicitar por teclado los valores para las variables rangoini y rangofin (rangofin debe ser mayor a rangini).
* Almacenar en el arreglo array valores aleatorios entre los rangos definidos por las variables rangoini y rangofin (Utilice la función rand ()).
* Imprima los valores almacenados en el arreglo.
* Solicite por teclado tres valores de tipo entero y asígnelos a la posición 3, 4 y 5 del arreglo respectivamente.
* Imprima el arreglo con los valores ingresados por teclado.
* Imprima las direcciones de los elementos en la posición inicial, 3, 4 y 5 respectivamente utilizando aritmética de punteros.

**¿Qué valores se almacena en el arreglo si se define un rango aleatorio entre 0 y 100? Ejecute varias veces el programa manteniendo el rango entre 0 y 100 y explique lo observado.**

**Utilice una función que no tenga retorno (void) cambiavalorpos(int\* arr); solicitar por teclado tres valores de tipo entero y asignarlos en las posiciones 3, 4 y 5 del arreglo utilizando aritmética de punteros, que antes hacía en el programa principal.**

* Imprima en pantalla la dirección en memoria de dicha función.

**IMPLEMENTACIONES Y RESULTADOS**

**EJERCICIO #1:**

[Texto del codigo]

[Imagen o imágenes que validen su correcta ejecución]

**EJERCICIO #2:**

[Texto del codigo]

[Imagen o imágenes que validen su correcta ejecución]

**CONCLUSIONES Y RECOMENDACIONES**

**BIBLIOGRAFIA**

[1]. Deitel and Deitel. Como programar en C/C++. Segunda edición o superior.

[2]. De la Fuente y otros. Aprenda lenguaje ANSI C como si estuviera en primero. Universidad de Navarra.

[3]. Instituto Nacional de Tecnologías Educativas y de Formación del Profesorado. Introducción a la programación con el lenguaje C. Enlace: http://recursostic.educacion.es/observatorio/version/v2/index.php?option=com\_content&view=article&id=745. Fecha de consulta: Viernes 2 de Marzo de 2018.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| **INICIAL** | **EN DESARROLLO** | **DESARROLLADO** | **EXCELENTE** |

**Firma del Profesor**