

Ingeniero en computación

Ingeniero en Software y tecnologías emergentes

Materia: Programación Estructurada / Clave **36276**

Alumno: Hernandez Ceseña Ivan Fernando

Matrícula: 373077

Maestro: Pedro Núñez Yépiz

Actividad No. : 10

Tema - Unidad : FUNCIONES y METODOS DE ORDENACION Y
BUSQUEDA ESTRUCTURAS Y LIBRERÍAS

Ensenada Baja California a 15 de octubre del 2023

1. INTRODUCCIÓN

Se van a usar funciones para tener un programa con un operador múltiple (switch) y poder acceder por medio de este a la acción que se quiere ejecutar. acompañados de ciclos para automatizar selecciones y entradas de datos. Utilizando vectores de cadena.

Se implementa el uso de una librería (.h) hecha por el alumno. En ella se encontrarán las funciones que se vayan a usar durante el programa.

También se implementa el método de búsqueda para poder llenar vectores de cadena sin repetir en este caso matrículas, que es por estas mismas por las que se manipula el registro de 500 espacios.

El otro método que se implementa es el de ordenamiento en base al método de la burbuja, pero implementando más funciones dentro del bloque de código de la función.

también agregamos un método de búsqueda extra que es el método de búsqueda binaria. Esto mismo nos sirve para que a la hora de tener nuestros registros ordenados no se realicen tantos procesos a la hora de buscar un usuario.

2. COMPETENCIA

usar funciones para hacer un código más claro fácil de entender, además de poner en práctica la optimización de código. utilizando vectores de cadenas, métodos de búsqueda y ordenación.

3. FUNDAMENTOS

tomado del manual de práctica

4. PROCEDIMIENTO

REALICE EL SIGUIENTE PROGRAMA QUE CONTENGA UN MENÚ.

MENÚ

- 1.- AGREGAR (AUTOM 10 REGISTROS)**
- 2.- AGREGAR MANUAL**
- 3- ELIMINAR REGISTRO (lógico)**
- 4.- BUSCAR**
- 5- ORDENAR**
- 6.- IMPRIMIR**
- 0.- SALIR**

UTILIZAR UN ARREGLO DE 500 REGISTROS

SE DEBERÁ UTILIZAR ESTRUCTURAS CON LOS DATOS BÁSICOS DE UN ALUMNO (status, Matricula, ApPat, ApMat, Nombre, Edad, Sexo)

Busqueda y Ordenacion por campo MATRICULA

5. RESULTADOS Y CONCLUSIONES

esta estructura genera un registro con diferentes tipos de datos.

```
#include "ivan.h"

#define N 500

typedef struct _alumnos{
    int status;
    int matricula;
    char apellidoPaterno[30];
    char apellidoMaterno[30];
    char nombre[30];
    int edad;
    int sexo; } Talumns;
```

en esta parte del codigo generamos un vector de este registro de datos

```
int op;
int posicion=0, bandera=0;
Talumns reg[N];
int numero = 10;
```

esto se hizo dentro de la función menú la cual controla todo nuestro codigo.

```

do{
    system("cls");
    msg();
    op= validacionNumero("elige una opcion", 0, 6);
    switch(op)
    {
        case 1:
            if((posicion + numero) > N)
            {
                numero = N - posicion;
            }
            registroAleatorio(reg, posicion, numero);
            imprimirAleatorio(reg, posicion, numero);
            printf("llenado de 50 registros exitoso\n");
            posicion= posicion + numero;
            bandera = 0;
            break;

        case 2:
            if(posicion + 1 > N)
            {
                printf("el registro esta lleno\n");
            }
            else
            {
                alumnoManual(reg, posicion);
                imprimirUnAlumno(reg, posicion);
                posicion= posicion+1;
            }
            break;

        case 3:
            eliminarRegistro(reg, posicion);
            break;

        case 4:
            buscarElemento(reg, posicion, bandera);
            break;

        case 5:
            ordenarAscendente(reg, posicion);
            bandera = 1;
            break;

        case 6:
            printf("registro\n");
            imprimirRegistro(reg, posicion);
            break;
    }
    system("pause");
}while(op!=0);

```

esta parte de la funcion menu es la que controla todas las acciones de nuestro programa.

Esta parte característica del código son los prototipos de todas las funciones que se usarán en el programa. tienen algo característico lo cual es que en todas las funciones que queramos manipular nuestro vector de registro tendremos que meter como parámetro al declarar nuestra función.

```
void registroAleatorio(Talumns reg[], int i, int numero);
void imprimirAleatorio(Talumns reg[], int posicion, int n);
void imprimirUnAlumno(Talumns reg[], int posicion);
void sexo(Talumns reg[], int i, int li, int ls);
void status(Talumns reg[], int i);
void matricula(Talumns reg[], int li, int ls, int i);
void apellidoPaterno(Talumns reg[], int i);
void apellidoMaterno(Talumns reg[], int i);
void nombre(Talumns reg[], int i);
void edad(Talumns reg[], int i, int li, int ls);
void alumnoManual(Talumns reg[], int i);
int existeValor(Talumns reg[], int n, int numero);
void eliminarRegistro(Talumns reg[], int posicion);
void buscarElemento(Talumns reg[], int posicion, int bandera);
void ordenarAscendente(Talumns reg[], int n);
int busquedaBinaria(Talumns reg[], int izquierda, int derecha);
void imprimirRegistro(Talumns reg[], int posicion);
```

El uso de una librería facilita mucho el entendimiento claro del código, también ayuda a que no se vea muy abultado. Esto nos ayuda a corregir errores de manera más rápida.

Los métodos de ordenamiento y búsqueda también son una herramienta bastante útil a la hora de usar funciones en las que se necesitan valores en órdenes específicos o que no se requieren valores repetidos.

El separar acciones específicas también ayuda a poder usarlas de mejor manera dentro de otros bloques de código y tener códigos más limpios y fáciles de entender.

6. ANEXOS

anexos en el otro archivo:

nombre del archivo: anexo_HCIF_RP10_PE

7. referencias

Diseño de algoritmos y su codificación en lenguaje C

Corona, M.A. y Ancona, M.A. (2011)..

España: McGraw-Hill.

ISBN: 9786071505712

Programación estructurada a fondo: implementación de algoritmos en C

:Pearson Educación.Sznajdleder, P. A. (2017)..

Buenos Aires,Argentina: Alfaomega

Como programar en C/C++

H.M. Deitel/ P.J. Deitel

Segunda edición

Editorial: Prentice Hall.

ISBN:9688804711

Programación en C. Metodología, estructura de datos y objetos

Joyanes, L. y Zahonero, I. (2001)..

España: McGraw-Hill.

ISBN: 8448130138