

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. : 9 3/4

Tema - Unidad : CADENAS.

Ensenada Baja California a 22 de octubre del 2023

1. INTRODUCCIÓN

Se van a usar funciones para tener un programa con un operador simple (if) 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 cadenas..

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.

2. COMPETENCIA

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

3. FUNDAMENTOS

- sintaxis de ciclos.
- funcionamiento de ciclos.
- proceso de ciclos.
- movilización de una función a otra.
- funcionamiento de vectores y matrices y cómo utilizarlas correctamente.
- funcionamiento de cadenas
- declaraciones de vectores y matrices.
- uso de librerías propias

4. PROCEDIMIENTO

Realiza un programa que sirva para generar el CURP de una persona.

El programa debe pedir los datos al usuario, generar, almacenar en una cadena y desplegar el CURP.

El programa deberá repetirse cuantas veces desee el usuario

Nota: el programa deberá estar 100% validado, de datos entrada así como las reglas que e deben cumplir al generar el curp

5. RESULTADOS Y CONCLUSIONES

en este programa se declaran prototipos de funciones que vamos a usar para mostrar menu's

```
int msg();
char primerVocal(char *cadena);
char egundaConsonante(char cadena[]);
int bisiesto(int anio);
int antisonante(char cadena[], char antisonantes[][5], int numAntisonantes);
void omitirNombres(char nombre[], char partes[][7]);
void omitirPartes(char apellido[], char partes[][6]);
```

en este caso la funcion principal controla nuestro programa

```
int main()
{
    srand(time(NULL));

    int op;
    char nombre[50];
    char apellidoPaterno[50];
    int opcion1;
    int opcion2;
    char apellidoMaterno[50];
    int dia;
    char Cdia[3];
    int mes;
    char Cmes[3];
    int anio;
    char canio[6];
    char gen;
    char est[3], estado;
    int ultimoDigito;
    char cultimoDigito[1];
    char curp[19];

    do{

        op = msg();

        if(op == 1)
        {
```

funciones para validacion de la curp y para leer datos

```
validacionTexto("dame el nombre", nombre, sizeof(nombre));
omitirNombres(nombre, nombresCortos);
omitirNombres(nombre, nombresCortos);
omitirPartes(nombre, partes);
omitirPartes(nombre, partes);
sinEspacio(nombre);

opcion1 = validacionNumero("tienes apellido paterno?\n1- si\n2- no", 1, 2);
if (opcion1 == 1)
{
    validacionTexto("dame tu apellido paterno", apellidoPaterno, sizeof(apellidoPaterno));
    omitirPartes(apellidoPaterno, partes);
    omitirPartes(apellidoPaterno, partes);
    sinEspacio(apellidoPaterno);
}

opcion2 = validacionNumero("tienes apellido materno?\n1- si\n2- no", 1, 2);
if (opcion2 == 1)
{
    validacionTexto("dame tu apellido materno", apellidoMaterno, sizeof(apellidoMaterno));
    omitirPartes(apellidoMaterno, partes);
    omitirPartes(apellidoMaterno, partes);
    sinEspacio(apellidoMaterno);
}

anio = validacionNumero("escribe tu anio de nacimiento (4 digitos)", 1910, 2023);
sprintf(canio, "%d", anio);

mes = validacionNumero("escribe tu mes de nacimiento", 1, 12);
sprintf(Cmes, "%d", mes);
```

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.

La lectura de todo como cadena facilita mucho su validación.

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_RP09_3_4_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