



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

Ingeniero en Software y tecnologías emergentes

Materia: Programación Estructurada / Clave 36276

Alumno: Antonio

Matrícula: 372576

Maestro: Pedro Núñez Yépiz

Actividad No. 9-3/4 : CURP

Tema - Unidad 5 : arreglos y Cadenas

Ensenada Baja California a 22 de octubre del 2023



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

1. INTRODUCCIÓN

Una cadena es una serie ordenada de caracteres, como letras, números y símbolos, que se almacena en la memoria de una computadora. Estos caracteres pueden representar texto o datos, y las cadenas se utilizan para manipular y procesar esta información en programas informáticos.

2. COMPETENCIA

La competencia que se busca para nosotros los alumnos es Fortalecer los conocimientos comprendidos en prácticas anteriores, como lo son las estructuras de control, los distintos tipos de ciclos y sus aplicaciones, así como los métodos de validación y las distintas funciones. Lo que se busca es seguir fortaleciendo los conocimientos sobre los arreglos y cadenas, centrándose principalmente en este punto, agregando distintos métodos de ordenación y búsqueda, así como iniciar con la creación de nuestra biblioteca, en la cual se encontraran las funciones usadas en prácticas anteriores y se agregaran las futuras funciones creadas.

3. FUNDAMENTOS

Las cadenas son colecciones de datos del mismo tipo, en este caso, datos de tipo char. Al querer acceder a algún dato de nuestra cadena se tiene que acceder mediante un índice el cual inicia en la posición 0 y termina en $n-1$.

La manera el cómo declaramos las cadenas es similar a cuando declaramos un vector colocando el tamaño de la cadena entre los corchetes;

Char cadena [n];

Para acceder a sus datos tenemos que tener en cuenta como mencioné anteriormente.

Por ejemplo si queremos acceder al segundo carácter de una cadena, esta al iniciar en 0 el segundo carácter no se encontraría en la posición 2, sino que en la posición 1.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

4. PROCEDIMIENTO

CURP

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

Instructivo:

<http://www.ordenjuridico.gob.mx/Federal/PE/APF/APC/SEGOB/Instructivos/InstructivoNormativo.pdf>

DATOS PARA PROBAR FUNCIONAMIENTO DE CURP

FRAIRE

LOPEZ

JOSE

OMAR

1997

31

ENERO

HOMBRE

GUERRERO

CURP: FXLO970131HGRRPM02

CHAN

CWEN

2002

MARZO

28

HOMBRE

NACIDO EN EL EXTRANJERO

CURP: CAXC020328HNEHXWA2

DE LA LOMA

XIU

JOSE

ANDRES

2007

DICIEMBRE

28

HOMBRE

YUCATAN

CURP: LOXA071228HYNMXNA3



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

5. RESULTADOS Y CONCLUSIONES

Las cadenas son un elemento importante en cualquier lenguaje de programación, almacenando información y datos que de otra manera no podríamos acceder a ellos. También la validación a pesar de ser casada de hacer es indispensable para el correcto funcionamiento del código

```
// Antonio Ramos Gonzalez Mt: 372576
// 10/12/2023 || 10/22/2023
// se pedirán datos al usuario para poder generar su CURP, haciendo todas las validaciones del mismo
// RGA_Act9.75_932

#include "Babilonia.h"

int msg();
void menu();
void curp();
int nombre(char curp[], char nomb[], char nomb2[], char apP[], char apM[]);
int nacimiento(char curp[]);
void sexo(char curp[]);
void estados(char curp[]);
void imprimirEstados();
void consonantes(char curp[], char nomb[], char nomb2[], char apP[], char apM[], int val);
void generacio(char curp[], int year);
void numAl(char curp[]);
int nomb_Novalid(char nomb[]);
bool palabrasInconvenientes(char curp[]);
void validConsoAp(char curp[], char cad[]);
void validConsoNomb(char curp[], char cad[]);
void validnumb(char cad[]);
void digitos(char curp[], char cad[]);
void caractapP(char curp[], char cad[]);
void caractapM(char curp[], char cad[]);
void caractano(char curp[], char cad[]);
void conv(char cad[]);

> int main() ...
> int msg() ...
> void menu() ...
> void curp() ...
> int nombre(char curp[], char nomb[], char nomb2[], char apP[], char apM[]) ...
```



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

```
140 > int nombre(char curp[], char nomb[], char nomb2[], char apP[], char apM[]) ...
204
205 > int nacimiento(char curp[]) ...
258
259 > void sexo(char curp[]) ...
278
279 > void estados(char curp[]) ...
297
298 > void imprimirEstados() ...
341
342 > void consonantes(char curp[], char nomb[], char nomb2[], char apP[], char apM[], int val) ...
384
385 > void generacio(char curp[], int year) ...
416
417 > void numAl(char curp[]) ...
429
430 > int nomb_Novalid(char nomb[]) ...
454
455 > bool palabrasInconvenientes(char curp[]) ...
552
553 > void validConsoAp(char curp[], char cad[]) ...
584
585 > void validConsoNomb(char curp[], char cad[]) ...
605
606 > void validnumb(char cad[]) ...
649
650 > void digitos(char curp[], char cad[]) ...
674
675 > void caractapP(char curp[], char cad[]) ...
704
705 > void caractapM(char curp[], char cad[]) ...
729
730 > void caractano(char curp[], char cad[]) ...
751
752 > void conv(char cad[]) ...
768
```

6. ANEXOS

RGA_RP09.75_PE (ANEXO)

GitHub: https://github.com/Anrago/Programacion-estructurada/tree/main/Actividad_9.75



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