



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

Ingeniero en computación

Ingeniero en Software y tecnologías emergentes

Materia: Programación Estructurada / Clave 36276

Alumno: Kevin Alejandro Gonzalez Torres

Matrícula: 372354

Maestro: Pedro Núñez Yépiz

Actividad Numero: 8

Tema - Unidad: Vectores y matrices

Ensenada Baja California a 1 de octubre del 2022



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

1. INTRODUCCIÓN

Esta práctica implica la creación de un programa en el lenguaje de programación C que tenga la capacidad de generar el Identificador Único de Registro de Población (CURP) de cualquier individuo al ingresar únicamente los datos esenciales, tales como nombre, apellidos, fecha de nacimiento, entidad federativa y sexo. Este programa debe estar equipado para manejar excepciones y verificar la validez de los datos ingresados, asegurando así la integridad y precisión de la CURP generada.

2. COMPETENCIA

El objetivo principal es aplicar conocimientos de programación en C, enfocándose en:

- 1.- Validar datos de entrada.
- 2.- Organizar el código de manera modular.
- 3.- Manejar excepciones y errores sin interrupciones inesperadas.

3. FUNDAMENTOS

Cadenas:

<https://www.programiz.com/c-programming/c-strings>

Structs:

<https://www.geeksforgeeks.org/structures-c/>

Funciones:

https://www.w3schools.com/c/c_functions.php

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 y deben cumplir al generar el curp.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

5. RESULTADOS Y CONCLUSIONES



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

```
170 void curp()
171 {
172
173     char nombre[50];
174     char primer_apellido[50];
175     int opcionApellido1;
176     int opcionApellido2;
177     char segundo_apellido[50];
178     int dia;
179     char cdia[3];
180     int mes;
181     char cmes[3];
182     int anio;
183     char canio[6];
184     char genero[2];
185     char entidad_nacimiento[3];
186     int ultimoDigito;
187     char cultimoDigito[1];
188     char curp[19];
189
190     printf("Ingrese los siguientes datos:\n");
191
192     printf("Nombre: ");
193     validateString(nombre, sizeof(nombre));
194     omitirNombres(nombre, nombresCortos);
195     omitirNombres(nombre, nombresCortos);
196     omitirPartes(nombre, partes);
197     omitirPartes(nombre, partes);
198     noSpacesString(nombre);
199     system("CLS");
200
201     printf("¿Tienes apellido paterno?\n");
202     printf("[1] Si\n[2] No\nOpcion: ");
203     opcionApellido1 = validate(1, 2);
204     if (opcionApellido1 == 1)
205     {
206         printf("Apellido Paterno: ");
207         validateString(primer_apellido, sizeof(primer_apellido));
208         omitirPartes(primer_apellido, partes);
209         omitirPartes(primer_apellido, partes);
210         noSpacesString(primer_apellido);
211         puts(primer_apellido);
212     }
213     system("CLS");
214
215     printf("¿Tienes apellido materno?\n");
216     printf("[1] Si\n[2] No\nOpcion: ");
217     opcionApellido2 = validate(1, 2);
218     if (opcionApellido2 == 1)
219     {
220         printf("Apellido Materno: ");
221         validateString(segundo_apellido, sizeof(segundo_apellido));
222         omitirPartes(segundo_apellido, partes);
223         omitirPartes(segundo_apellido, partes);
224         noSpacesString(segundo_apellido);
```



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

```
228 printf("Año: ");
229 anio = validate(1910, 2023);
230 sprintf(canio, "%d", anio);
231 system("CLS");
232
233 printf("Mes: ");
234 mes = validate(1, 12);
235 sprintf(cmes, "%d", mes);
236 system("CLS");
237
238 printf("Día: ");
239 if (mes == 2 && esBisiesto(anio) == 1)
240 {
241     dia = validate(1, 29);
242 }
243 else
244 {
245     if (mes == 2 && esBisiesto(anio) == 0)
246     {
247         dia = validate(1, 28);
248     }
249     else
250     {
251         if (mes == 1 || mes == 3 || mes == 5 || mes == 7 || mes == 8 || mes == 10 || mes == 12)
252         {
253             dia = validate(1, 31);
254         }
255         else
256         {
257             if (mes == 4 || mes == 6 || mes == 9 || mes == 11)
258             {
259                 dia = validate(1, 30);
260             }
261         }
262     }
263 }
264 sprintf(cdía, "%d", dia);
265 system("CLS");
266
267 printf("Género (H o M): ");
268 validateString(genero, sizeof(genero));
269 while (genero[0] != 'H' && genero[0] != 'M')
270 {
271     printf("Error - Letra no aceptada, solo H o M.\n");
272     printf("Género (H o M): ");
273     validateString(genero, sizeof(genero));
274 }
275 system("CLS");
```



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

```
311 printf("Entidad de Nacimiento: ");
312 validateString(entidad_nacimiento, sizeof(entidad_nacimiento));
313 while (validarEntidad(entidad_nacimiento, estados, sizeof(estados) / sizeof(estados[0])) == 0)
314 {
315     printf("Entidad no válida. Ingresar una abreviatura válida.\n");
316     printf("Entidad de Nacimiento: ");
317     validateString(entidad_nacimiento, sizeof(entidad_nacimiento));
318 }
319 system("CLS");
320
321 if (opcionApellido1 == 1)
322 {
323     curp[0] = toupper(primer_apellido[0]);
324     curp[1] = obtenerPrimeraVocal(primer_apellido);
325 }
326 else
327 {
328     if (opcionApellido1 == 2)
329     {
330         curp[0] = 'X';
331         curp[1] = 'X';
332     }
333 }
334 if (opcionApellido2 == 1)
335 {
336     curp[2] = toupper(segundo_apellido[0]);
337 }
338 else
339 {
340     if (opcionApellido2 == 2)
341     {
342         curp[2] = 'X';
343     }
344 }
345
346 curp[3] = toupper(nombre[0]);
347 curp[4] = canio[2];
348 curp[5] = canio[3];
349 if (mes < 10)
350 {
351     curp[6] = '0';
352     curp[7] = cmes[0];
353 }
354 else
355 {
356     curp[6] = cmes[0];
357     curp[7] = cmes[1];
358 }
359 if (dia < 10)
360 {
361     curp[8] = '0';
362     curp[9] = cdia[0];
363 }
```



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

```
364     else
365     {
366         curp[8] = cdia[0];
367         curp[9] = cdia[1];
368     }
369
370     curp[10] = toupper(genero[0]);
371     curp[11] = entidad_nacimiento[0];
372     curp[12] = entidad_nacimiento[1];
373     if (!obtenerSegundaConsonante(primer_apellido) || opcionApellido1 == 2)
374     {
375         curp[13] = 'X';
376     }
377     else
378     {
379         curp[13] = obtenerSegundaConsonante(primer_apellido);
380     }
381     if (!obtenerSegundaConsonante(segundo_apellido) || opcionApellido2 == 2)
382     {
383         curp[14] = 'X';
384     }
385     else
386     {
387         curp[14] = obtenerSegundaConsonante(segundo_apellido);
388     }
389
390     if (!obtenerSegundaConsonante(nombre))
391     {
392         curp[15] = 'X';
393     }
394     else
395     {
396         curp[15] = obtenerSegundaConsonante(nombre);
397     }
398     if (anio < 2000)
399     {
400         curp[16] = '0';
401     }
402     else
403     {
404         if (anio >= 2000 && anio < 2010)
405         {
406             curp[16] = 'A';
407         }
408         else
409         {
410             if (anio >= 2010)
411             {
412                 curp[16] = 'B';
413             }
414         }
415     }
416     ultimoDigito = randomNumber(0, 9);
417     sprintf(cultimoDigito, "%d", ultimoDigito);
418     curp[17] = cultimoDigito[0];
419     curp[18] = '\0';
```



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

```
417     sprintf(cultimoDigito, "%d", ultimoDigito);
418     curp[17] = cultimoDigito[0];
419     curp[18] = '\\0';
420
421     if (validarAntisonante(curp, antisonantes, sizeof(antisonantes) / sizeof(antisonantes[0])))
422     {
423         curp[1] = 'X';
424     }
425
426     printf("La CURP generada es: %s\\n", curp);
427 }
428
```

```
1  #include "kewvyValidates.h"
2  #include "listas.h"
3
4  int msges();
5  void menu();
6  char obtenerPrimeraVocal(const char cadena[]);
7  char obtenerSegundaConsonante(char cadena[]);
8  int esBisiesto(int anio);
9  int validarEntidad(char cadena[], char entidades[][3], int numEntidades);
10 int validarAntisonante(char cadena[], char antisonantes[][5], int numAntisonantes);
11 void curp();
12
```

6. ANEXOS

GTKA_RP09_PE_ANEXOS



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

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