

Actividad | 3 | Calculo de RFC

Lenguajes de Programación I

Ingeniería en Desarrollo de Software



TUTOR: Francisco Ortega Rivera

ALUMNO: Ana Laura Reyes Rios

FECHA: 19 de Abril del 2025

Índice

1. Introducción.....	3
2. Descripción.....	4
3. Justificación.....	5
4. Desarrollo.....	6
4.1 Codificación.....	6
4.2 Prueba del sistema.....	6
5. Conclusión.....	7
6. Referencias.....	8

1. Introducción

En el siguiente documento vamos a presentar el tema de dicha actividad que es realizar un programa para calcular el RFC sin la homoclave el cual lo realizaremos en el lenguaje C++ como ya hemos mencionado en la anteriores actividades este lenguaje de programación tiene como propósito en general que se utilice principalmente para el desarrollo de software de sistemas, aplicaciones de escritorio, juegos y software de alto rendimiento, entre otras aplicaciones, este programa se trata de una extensión del lenguaje de programación C, por lo que hereda muchas de sus características. Como ya lo hemos dicho anteriormente este programa es conocido especialmente por su eficiencia, velocidad y capacidad para trabajar a bajo nivel. Pasando a nuestra actividad donde vamos a presentar dicho código para realizar el cálculo del RFC con las condiciones que nos solicita, para ello nos guiaremos con las indicaciones que nos exige en esta misma actividad.

2. Descripción

A continuación daremos una explicación de lo que realizaremos en dicha actividad ya que se nos presenta realizar un programa en el cual se nos pide realizar un código con dichas condiciones para calcular el RFC sin homoclave el cual lo crearemos en el lenguaje C++ para demostrar dicho código que funciona se capturaran pantallas y se agregaran en el apartado del desarrollo con otros dos apartados el cual será uno llamado codificación y el otro prueba de sistema.

Para complementar dicha actividad realizaremos los demás apartados requeridos, como en la parte principal nuestra portada para después seguir con el índice el cual mostrara la estructura de nuestra actividad, el siguiente es la introducción donde daremos una presentación del tema de esta actividad, seguiremos con la descripción la cual vamos a interpretar con nuestras propias palabras el contexto y lo solicitado en dicha actividad, sigue la justificación donde explicaremos por que debería emplearse dicha solución en esta actividad y como ya mencionamos antes se pondrá la codificación de dicho programa en el desarrollo explicando que es lo que sucede cuando lo empleamos, por penúltimo redactaremos un conclusión de dicha actividad explicando la importancia de aprender dicho tema y por ultimo agregaremos las referencias utilizadas para la investigación de dicha actividad.

3. Justificación

En este apartado explicaremos por que empleamos este tipo de solución recordando lo que explicamos en la actividad anterior, emplear dicho lenguaje para codificar operaciones básicas es lo que nos requiere realizar por ello es importante repasar la utilidad de dicho programa el cual hay que comprender y entender dicho lenguaje, ya que constituye en una opción popular para una amplia gama de aplicaciones de software en diversos sectores, desde sistemas operativos hasta aplicaciones empresariales y de entretenimiento. Entre las razones en las cuales radica el gran impacto en el ámbito tecnológico, se destacan las siguientes: Su eficiencia y capacidad para trabajar a bajo nivel, hace ideal para aplicaciones que requieren un alto rendimiento y uso eficiente de los recursos del sistema, es portable que puede ejecutarse en una variedad de plataformas y sistemas operativos, siendo muy versátil para el desarrollo de software multiplataforma, es ampliamente utilizado en la industria del software y tiene una gran base de usuarios y bibliotecas de código abierto disponibles, el cual facilita el desarrollo de software y la colaboración entre desarrolladores y es un lenguaje orientado a objetos, permitiendo la encapsulación, la herencia y el polimorfismo, lo que facilita la creación de software modular, reutilizable y fácil de mantener.

4. Desarrollo

4.1 Codificación

En este apartado presentamos el código realizado el cual nos pide realizar un RFC sin homoclave para ello se siguió las mismas indicaciones que el profesor y se presentan la captura de pantalla de cada uno de los pasos realizados para llegar al resultado deseado tome prácticamente su mismo código para ir viendo cómo se iba realizando poniendo atención a cada una de las líneas del código.

```
1- /*****
2- Autor: Ana Laura Reyes
3- Actividad 3. Programa en C++ que realiza el calculo del RFC de acuerdo con Las
4- reglas de SAT
5- Fecha: 19/04/2025
6-
7- *****/
8- #include <iostream> //biblioteca estandar para entradas y salidas en C++
9- #include <string> //biblioteca para manipulacion de cadenas (std::string)
10-
11- // funcion para obtener la primera vocal interna del apellido paterno
12- char obtenerPrimeraVocalInterna (const std::string& str) {
13-     //Recorremos la cadena desde el segundo caracter (indice 1) hasta el final
14-     for (size_t i=1; i < str.length(); ++i) {
15-         char c = str[i]; // se obtiene el caracter actual de la cadena
16-
17-         //se verifica si el carater es una vocal mayuscula
18-         if (c == 'A' || c == 'E' || c == 'I' || c == 'O' || c == 'U')
19-             return c; // Regresa la primera vocal interna encontrada
20-     }
21-     return 'X'; //si no se encuentra ninguna vocal interna, se devuelve una X
22- }
```

```
60 }
61
62 int main(){
63     std::string nombre, apellidoPaterno, apellidoMaterno, fechaNacimiento;
64
65     // solicita el nombre del usuario
66     std::cout << "Ingrese el nombre: ";
67     std::getline(std::cin, nombre); // captura toda la línea del nombre
68
69     // solicita el apellido paterno
70     std::cout << "Ingrese el apellido paterno: ";
71     std::getline(std::cin, apellidoPaterno);
72
73     // solicita el apellido materno, permitiendo que el usuario deje vacío
74     std::cout << "Ingrese el apellido materno (Si no tiene, presione Enter): ";
75     std::getline(std::cin, apellidoMaterno);
76
77     // solicita la fecha de nacimiento en formato YYYY-MM-DD
78     std::cout << "Ingrese la fecha de nacimiento (YYYY-MM-DD): ";
79     std::getline(std::cin, fechaNacimiento);
80
81     // se calcula el RFC utilizando la funcion calcularRFC
82     std::string rfc = calcularRFC(nombre, apellidoPaterno, apellidoMaterno, fechaNacimiento);
83
84     // se muestra el RFC generado
85     std::cout << "RFC: " << rfc << std::endl;
86
87     return 0; // finaliza el Programa
88 }
```

```
23 // funcion para calcular el RFC con base en los datos personales
24 std::string calcularRFC(const std::string& nombre, const std::string& apellidoPaterno, const std::string& apellidoMaterno, const std::string& fechaNacimiento){
25     std::string rfc; //variable para almacenar el RFC resultante
26
27     // se obtiene la primera letra del apellido paterno
28     char letraInicial = apellidoPaterno[0];
29
30     // se obtiene la primera vocal interna de apellido paterno
31     char PrimeraVocalInterna = obtenerPrimeraVocalInterna(apellidoPaterno);
32
33     // se obtiene la inicial del apellido materno, si no existe se usa x
34     char inicialApellidoMaterno = (!apellidoMaterno.empty()) ? apellidoMaterno[0] : 'X';
35
36     // se obtiene la inicial del nombre
37     char inicialNombre = nombre[0];
38
39     // se obtiene los ultimos dos digitos del año de nacimiento (posicion 2 y 3)
40     std::string anio = fechaNacimiento.substr(2,2);
41
42     // se obtiene el mes de nacimiento (posicion 5 y 6)
43     std::string mes = fechaNacimiento.substr(5,2);
44
45     // se obtiene el día de nacimiento (posicion 8 y 9)
46     std::string dia = fechaNacimiento.substr(8,2);
47
48     // se contruye el RFC conectando los valores obtenidos
49     rfc = letraInicial;
50     rfc += PrimeraVocalInterna;
51     rfc += inicialApellidoMaterno;
52     rfc += inicialNombre;
53     rfc += anio;
54     rfc += mes;
55     rfc += dia;
56
57     return rfc; // se retorna el RFC generado
58 }
```

4.2 Prueba del sistema

RFC CON APELLIDO MATERNO

```
Ingrese el nombre: JESUS
Ingrese el apellido paterno: VEGA
Ingrese el apellido materno (Si no tiene, presione Enter): CASTRO
Ingrese la fecha de nacimiento (YYYY-MM-DD): 1988-03-26
RFC: VECJ880326
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

```
Ingrese el nombre: ANA
Ingrese el apellido paterno: REYES
Ingrese el apellido materno (Si no tiene, presione Enter): RIOS
Ingrese la fecha de nacimiento (YYYY-MM-DD): 1988-03-18
RFC: RERA880318
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

RFC SIN APELLIDO MATERNO

```
Ingrese el nombre: JESUS
Ingrese el apellido paterno: VEGA
Ingrese el apellido materno (Si no tiene, presione Enter):
Ingrese la fecha de nacimiento (YYYY-MM-DD): 1988-03-26
RFC: VEXJ880326
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

```
Ingrese el nombre: ANA
Ingrese el apellido paterno: REYES
Ingrese el apellido materno (Si no tiene, presione Enter):
Ingrese la fecha de nacimiento (YYYY-MM-DD): 1988-03-18
RFC: REXA880318
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

5. Conclusión

En esta actividad aprendimos como realizar un código para calcular el RFC para ello vimos fuentes para poder realizar dicha actividad y poder comprenderla correctamente, es por eso que también daremos un repaso de todos lo que vimos desde esta actividad hasta la anteriores por ello recordaremos que este es un lenguaje está enfocado en la programación orientada a objetos, la programación genérica y la programación de procedimientos. Se trata de una evolución del lenguaje C, que incorpora importantes mejoras y características adicionales. Dicho lenguaje es excepcionalmente útil en proyectos que requieren de un alto rendimiento y eficiencia en el uso de los recursos, como videojuegos, aplicaciones de realidad virtual, motores gráficos, y software de sistemas. Permiten manipular datos, realizar cálculos, tomar decisiones y controlar el flujo de la ejecución, lo que es esencial para el desarrollo de cualquier aplicación. Con esto nos pudimos dar cuenta que cuando formulamos el código deseado manipulamos los datos para poder obtener el resultado requerido en esta actividad.

6. Referencias

Workana, & Workana. (2021, 5 abril). C++: Qué es, Para qué sirve, Ventajas y Desventajas. Glosario - Workana | el Glosario Workana Explica Terminología del Mundo Freelance, Conceptos Fundamentales del Marketing y los Negocios. <https://i.workana.com/glosario/que-es-c/>

Moreno, P. (2024, febrero 13). ¿Qué es el lenguaje C++ y para qué sirve? Centro de e-Learning UTNBA. <https://blog.centrodelearning.com/lenguaje-c/>

(s.f.). Universitat de València. <https://www.uv.es/~sto/cursos/c++/curso95.pdf>

Equipo editorial de IONOS. (2023, 30 agosto). Los operadores de C++ de un vistazo: los operadores más importantes. IONOS Digital Guide. <https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/c-operators/>

TylerMSFT. (s. f.). Operadores de asignación. Microsoft Learn. <https://learn.microsoft.com/es-es/cpp/cpp/assignment-operators?view=msvc-170>

C++ Variables de tipo int y sus operaciones básicas | Aprende Programación Competitiva. (s. f.). <https://aprende.olimpiada-informatica.org/cpp-int-operaciones>

De Contenidos de GoDaddy, E. (2024, 22 enero). Programación en C++: Tips y mejores prácticas. GoDaddy Resources - LATAM. <https://www.godaddy.com/resources/latam/stories/programacion-en-c-tips-y-mejores-practicas>

Robledano, A. (2024, 4 septiembre). Qué es C++ y su importancia en la programación moderna. OpenWebinars.net. <https://openwebinars.net/blog/que-es-cpp/#:~:text=Una%20de%20las%20caracter%C3%ADsticas%20m%C3%A1s,clave%20de%20la%20POO%20como:>

Enlace para acceder a Github <https://github.com/AnaRR8803/Trabajos-Universitarios-18.git>