

Actividad [#3] - [Aplicación para cálculo de Rfc]

[Lenguajes de Programación 1]

Ingeniería en Desarrollo de Software

Tutor: Miguel Ángel Rodríguez Vega

Alumno: Ricardo Rivas Rocha

Fecha:05-Marzo-2023

Índice

Portada Página 1

Índice Página 2

Introducción Página 3

Descripción Página 4

Justificación Página 5

Investigación Página 6

Desarrollo Página 7 a 9

- Codificación
- Prueba del Sistema

Conclusión Página 10

Referencias Página 11

Introducción

En este trabajo realizaremos una aplicación que nos ayude básicamente a calcular un rfc de un nuevo empleado dentro de la empresa Coppel y pues solo necesitaríamos su información de su nombre completo con apellidos y fecha de nacimiento para calcular este mismo en sistema.

Descripción

En esta actividad se nos pide que usemos variables que en este caso van a hacer tipo enteros, algunos strings y arreglos esto para que nuestro código o aplicación lo pueda ejecutar en base a lo que hemos aprendido en estas 2 ultimas Actividades pasadas haciendo un sistema funcional a la hora de pedir los datos al usuario dentro de este mismo y solucionando búsquedas exhaustivas en internet u otro medio.

Justificación

En este pequeño programa que realizamos nos va a servir para aprender conceptos de la programación básica que es y cómo usar variables, así como también a usar funciones y hacer funciones fáciles de igual manera también orientadas a los objetos para una organización o empresa que requiera de algún desarrollo de alguna aplicación de Software en base a sus requerimientos y limitaciones y así poder saber que es lo que necesitan en base al análisis y diseño de este mismo.

Investigación

¿Qué es un Objeto en C++?

Un objeto (object en Ingles) es “un ente con su estado” (an entity with a state). Esto quiere decir que un objeto no solo es data inerte, sino que tiene la capacidad de cambiar las condiciones en que existe y por tanto como se comporta. Por ejemplo, una flor puede ser representada por una abstraccion compuesta de solo datos, es decir: variedad, color, peso, longevidad, etcetera, usando una variable struct en C.

Pero también puede ser representada por una abstraccion objeto la cual incluye “variables describiendo sus condiciones de existencia” (state variables) como, por ejemplo: edad, salud (Viva, muerta, marchita, enferma, podrida), y metodos (methods) como envejecer, crecer, morir, enfermar que cuando expresados en un mensaje a un objeto de tipo “Flor” este responde con un comportamiento.

Entonces, si a un objeto flor de tres dias de edad, que está Viva y con una longevidad de 12 dias, le mandamos el mensaje “envejece (14 dias)” el objeto flor cambia su estado a “muerta”. Un segundo mensaje TomaAgua(3 gotas) resulta en el comportamiento de no absorber nada de agua puesto que está muerta.

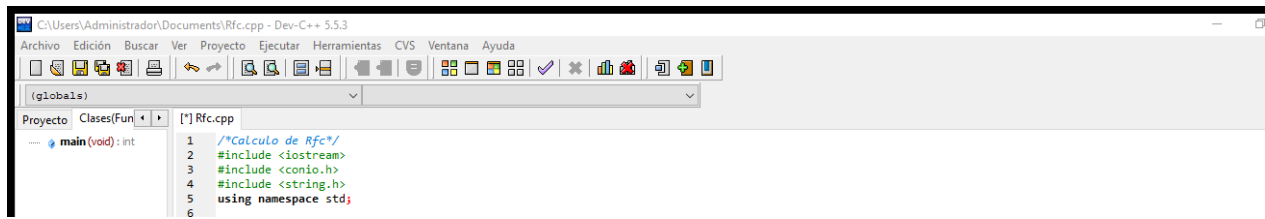
La palabra “objeto” en la metalanguage de C se refiere solamente a una porcion de memoria con nombre o anonima. Puesto que todos los objetos en C++ ocupan memoria, se podría decir que estos son una elaboracion mas compleja, que añaden a la entidad para agrupar data del lenguaje C (struct) lo siguiente: 1. Variables de estado (state variables), 2. Metodos, 3. Mensajes aceptables, y 4. Comportamientos.

Todo esto parece complejidad innecesaria hasta que uno efectua la abstraccion objeto de una impresora. La simplificación que resulta es enorme. Yo he escrito programas con orientación de objetos que son hasta diez veces más cortos que sus equivalentes con abstraccion de data y funciones (especialmente cuando se usa inheritance). Pero aún más importante es que programas diseñados con abstracciones objetos son mucho mas faciles de extender, integrar y corregir (debug).

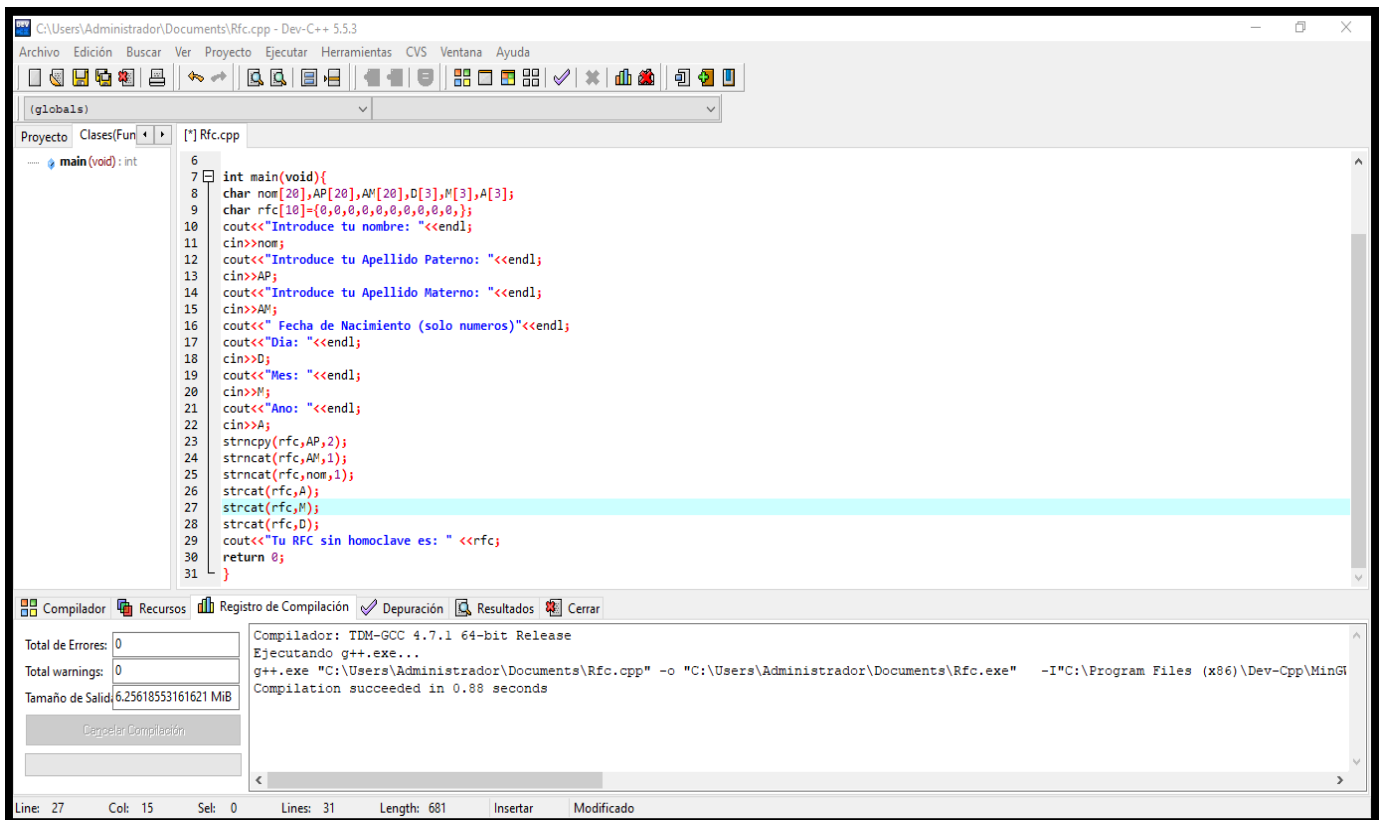
¿Qué es una clase en C++?

Las clases de C++ son, de forma predeterminada, tipos de valor. Se pueden especificar como tipos de referencia, lo que permite el comportamiento polimórfico para admitir la programación orientada a objetos. A veces, los tipos de valor se ven desde la perspectiva del control de la memoria y el diseño, mientras que los tipos de referencia se refieren a clases base y funciones virtuales con fines polimórficos. De forma predeterminada, los tipos de valor se pueden copiar, lo que significa que siempre hay un constructor de copias y un operador de asignación de copia. Para los tipos de referencia, haga que la clase no se puede copiar (deshabilite el constructor de copia y el operador de asignación de copia) y use un destructor virtual, que admite su polimorfismo previsto. Los tipos de valor están también relacionados con los contenidos que, cuando se copian, le proporcionan siempre dos valores independientes que se pueden modificar por separado. Los tipos de referencia están relacionados con la identidad: ¿qué tipo de objeto es? Por este motivo, los "tipos de referencia" también se conocen como "tipos polimórficos".

Codificación



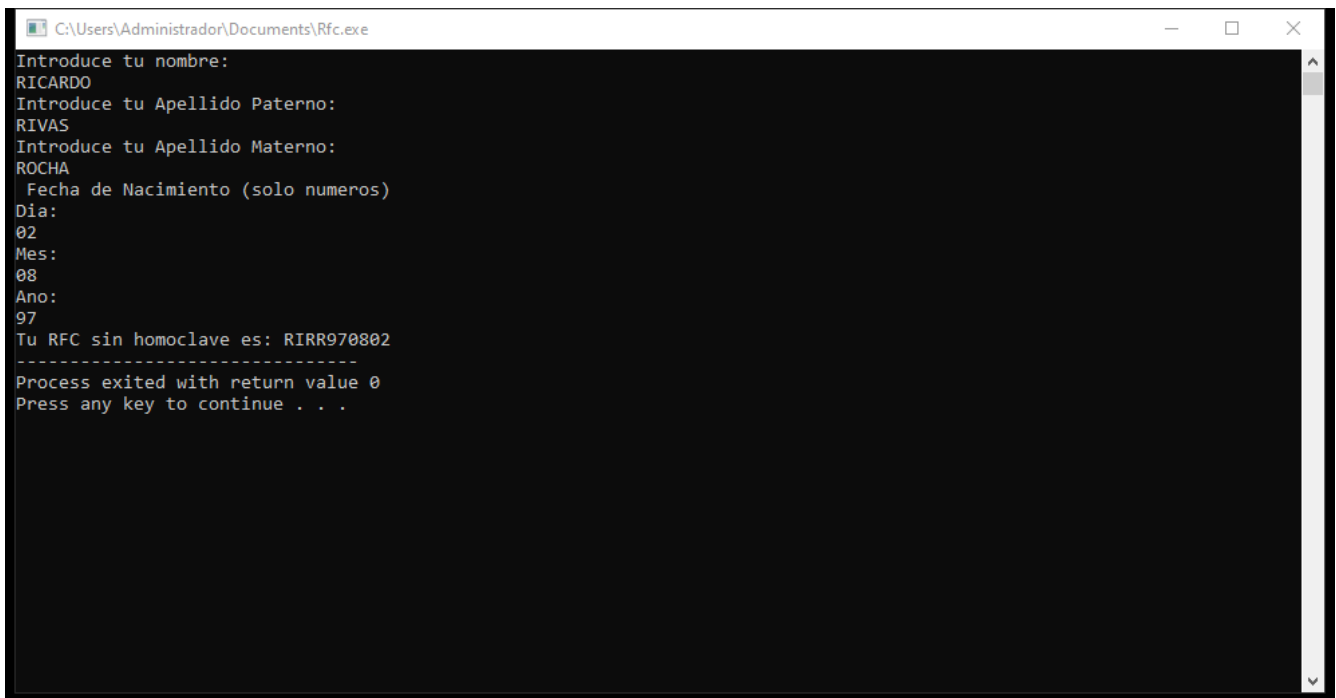
En esta primera imagen mando a llamar a las librerías de entrada salida la conio.h y la string porque voy a trabajar con cadenas de datos y enseguida les muestro el código de la aplicación que realice.



En esta ocasión creo mi espacio de trabajo que es int main y dentro de este declaro mis variables de tipo texto con char y les indico cuantos valores pueden almacenar de mis datos en seguido creo otra variable igualmente tipo char que sea mi rfc le indico los valores a almacenar que son 10 y para que no me de errores de escritura a esta le agrego ceros a los 10 valores indicados enseguida empiezo a crear mis impresiones de pantallas y guardarlas en sus

mismas variables al final de todo esto llamo a una función que se llama `strncpy` que me va a ayudar a copiar los primeros valores de mi rfc que son las 2 primeras letras del apellido paterno y la primera letra del apellido paterno y la inicial del nombre indicándole cuantas palabras quiero que me de y enseguida mando a llamar al año de nacimiento mes y día con la función `strcat` que me va a copiar los valores que les di a esas variables y al final imprimo y guardo en variable rfc el dato que requiere mi programa y finalizo con `return 0`.

Prueba del Sistema



```
C:\Users\Administrador\Documents\Rfc.exe
Introduce tu nombre:
RICARDO
Introduce tu Apellido Paterno:
RIVAS
Introduce tu Apellido Materno:
ROCHA
Fecha de Nacimiento (solo numeros)
Dia:
02
Mes:
08
Año:
97
Tu RFC sin homoclave es: RIRR970802
-----
Process exited with return value 0
Press any key to continue . . .
```

Aquí en esta imagen se muestra el proceso que hace mi código mostrando los datos requeridos por la actividad numero 3 ya sé que no es el mejor y nunca supe como ponerle la ñ y por que no le puse los 4 dígitos poco a poco se ira mejorando esta vaina jejeje el chiste es que si corrió y pues estoy contento porque si quedo mi rfc sin homoclave el chiste es aprender.

Conclusión

Bueno primero que nada agradezco que tengamos este tipo de materias por que me gustan mucho las computadoras y que nos enseñen sus lenguajes de estas ya pasamos por el área del proceso del algoritmo diagrama de flujo y codificación y en la anterior materia nos dedicamos al análisis de estos mismos programas y con la ayuda de las matemáticas podemos hacer iteraciones con la computadora para que se nos haga más fácil entender a esta ya se que el camino no ha sido fácil hasta este momento pero con esfuerzo y dedicación se pueden conseguir esos logros que todos anhelamos en nuestras vidas y uno de esos es tener mejor estabilidad económica aprender y aprender por que se van renovando los programas computacionales diferentes lenguajes librerías etc. Quedo muy satisfecho con lo que he aprendido ahora por que cuando hacemos el trabajo cada quien ahí vemos que estamos hechos y con la ayuda de los tutores seamos mejores aún más.

Referencias

(Rodolfo Martinez III, 2019)
(Microsoft, 2023)

Referencias

Microsoft. (03 de 03 de 2023). *Learn-Microsoft.com*. Obtenido de <https://learn.microsoft.com/es-es/cpp/cpp/value-types-modern-cpp?view=msvc-170>
Rodolfo Martinez III. (10 de 01 de 2019). *Quora*. Obtenido de <https://es.quora.com/Qu%C3%A9-son-los-objetos-en-C-y-C++-En-qu%C3%A9-manera-son-diferentes>