

Actividad 3 - Lenguaje de Programación 1

Ingeniería en Desarrollo de Software

Tutor: Francisco Ortega

Alumno: José Luis Rodríguez Blancas

Fecha: 26/02/2023

índice

Introducción.....	3
Descripción.....	4
Justificación.....	5
Desarrollo.....	6
Conclusión.....	8
Referencias.....	9

Introducción

En el contexto de la constructora AMC, se plantea la necesidad de desarrollar un programa en lenguaje C++ que permita calcular el RFC (Registro Federal de Contribuyentes) de los nuevos empleados. El RFC es un identificador fiscal utilizado en México y se forma a partir de información clave, como el nombre, apellidos y fecha de nacimiento del individuo.

El programa que se busca crear se basa en una serie de reglas que siguen el formato estándar para la generación del RFC. Para lograrlo, se hará uso de los conceptos de Objetos y Clases, conocimientos fundamentales en la materia de programación. La tarea consiste en capturar los datos esenciales de cada empleado, procesarlos y obtener el RFC resultante según las reglas establecidas.

El RFC se compone de diferentes elementos que reflejan información específica del individuo, como las iniciales de sus apellidos y nombre, así como su fecha de nacimiento. Es importante destacar que en esta aplicación no se generará la homoclave, ya que esta es proporcionada por el SAT (Servicio de Administración Tributaria) a través de un proceso particular.

El objetivo de esta actividad es proporcionar una herramienta funcional y automatizada que simplifique el proceso de obtención del RFC para los nuevos empleados de la constructora AMC. Mediante la implementación de este programa en C++, se espera mejorar la eficiencia en el registro y gestión de la información fiscal de los trabajadores, garantizando así el cumplimiento legal y tributario de la empresa.

Descripción

En el contexto presentado, se describe la necesidad de desarrollar un programa informático que permita calcular el RFC de los nuevos empleados de la constructora AMC en México. El RFC (Registro Federal de Contribuyentes) es un identificador fiscal esencial para cualquier individuo o empresa que realice actividades económicas en el país. Para obtenerlo, se requiere capturar ciertos datos personales, como el nombre, apellido paterno, apellido materno y fecha de nacimiento de cada empleado.

La actividad propuesta consiste en crear una aplicación en lenguaje C++ utilizando los conceptos de Objetos y Clases, que son fundamentales en la programación orientada a objetos. Estos conceptos nos permiten organizar y estructurar el código de manera más eficiente y modular, facilitando el proceso de desarrollo y mantenimiento del programa.

Para generar el RFC, se deben seguir ciertas reglas específicas, que involucran la combinación de las iniciales de los apellidos y del primer nombre, así como la inclusión de la fecha de nacimiento en las posiciones correspondientes del RFC. Es importante mencionar que en esta actividad no se generará la homoclave, ya que esta es asignada por el SAT a través de un proceso que utiliza un software alfanumérico.

El propósito de esta actividad es brindar una herramienta automatizada que agilice el proceso de cálculo del RFC para los empleados de la constructora AMC, garantizando su correcta identificación y cumplimiento de obligaciones fiscales. Al utilizar el lenguaje de programación C++ y aplicar los conceptos de Objetos y Clases, se busca lograr una solución eficiente y bien estructurada para resolver el problema planteado.

Justificación

La elección de emplear una solución basada en los conceptos de Objetos y Clases en lenguaje C++ para la actividad de cálculo del RFC de los nuevos empleados de la constructora AMC es justificada por varias razones fundamentales.

En primer lugar, el enfoque de programación orientada a objetos proporciona una manera organizada y estructurada de abordar el problema. Al utilizar clases, se puede encapsular la lógica relacionada con el RFC en un objeto, lo que facilita la reutilización del código y el mantenimiento a largo plazo. Además, el paradigma de objetos permite modelar las entidades del problema de manera más natural, como la representación de un "Empleado" con atributos como nombre, apellidos y fecha de nacimiento, y métodos para calcular el RFC.

En segundo lugar, el lenguaje C++ ofrece ventajas como la eficiencia en el uso de recursos y la posibilidad de trabajar con punteros y estructuras de datos complejas. Esto es relevante cuando se manejan múltiples registros de empleados y se necesitan estructuras de datos más avanzadas para almacenar y manipular la información.

Además, la legibilidad y claridad del código se ven favorecidas por el enfoque orientado a objetos en comparación con otros paradigmas de programación, lo que facilita la comprensión y colaboración entre desarrolladores.

En resumen, emplear una solución basada en Objetos y Clases en C++ para calcular el RFC de los empleados de la constructora AMC permitirá una implementación organizada, eficiente y fácil de

mantener, al mismo tiempo que mejorará la legibilidad y comprensión del código por parte de los desarrolladores involucrados en el proyecto.

Desarrollo

Codificación:

The screenshot displays the OnlineGDB web interface in a browser. The main editor shows a C++ program named `main.cpp` with the following code:

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 class Empleado {
6 private:
7     string nombre;
8     string apellidoPaterno;
9     string apellidoMaterno;
10    string fechaNacimiento;
11
12 public:
13    void CapturarDatos() {
14        cout << "Ingrese el nombre: ";
15        cin.ignore();
16        getline(cin, nombre);
17
18        cout << "Ingrese el apellido paterno: ";
19        getline(cin, apellidoPaterno);
20
21        cout << "Ingrese el apellido materno (dejar en blanco si no hay): ";
22        getline(cin, apellidoMaterno);
23
24        cout << "Ingrese la fecha de nacimiento (en formato dd/mm/aaaa): ";
25        getline(cin, fechaNacimiento);
26    }
27 }
```

The interface includes a left sidebar with navigation links (IDE, My Projects, Classroom, Learn Programming, Programming Questions, Sign Up, Login) and a right sidebar with debugging tools (Call Stack, Local Variables, Display Expressions, Breakpoints and Watchpoints). The bottom status bar shows the system temperature (21°C), location (Parc. soleado), and the time (06:05 p.m. on 26/07/2023).

OnlineGDB beta
online compiler and debugger for c/c++

code. compile. run. debug. share.

IDE
My Projects
Classroom **new**
Learn Programming
Programming Questions
Sign Up
Login

GOT AN OPINION?
SHARE AND GET REWARDED.
Rakuten AIP

Have fun taking surveys and get paid!

About • FAQ • Blog • Terms of Use • Contact
Us • GDB Tutorial • Credits • Privacy
© 2016 - 2023 GDB Online

SPONSOR Auth0 — Implement Auth0 in any app in just 5 minutes

main.cpp

```

25  getline(cin, fechaNacimiento);
26  }
27
28  string CalcularRFC() {
29      string rfc;
30
31      rfc += apellidoPaterno[0];
32
33      for (size_t i = 1; i < apellidoPaterno.length(); i++) {
34          if (apellidoPaterno[i] == 'A' || apellidoPaterno[i] == 'E' ||
35              apellidoPaterno[i] == 'I' || apellidoPaterno[i] == 'O' || apel
36              rfc += apellidoPaterno[i];
37              break;
38          }
39      }
40
41      rfc += (apellidoMaterno.empty()) ? 'X' : apellidoMaterno[0];
42      rfc += (nombre.empty()) ? 'X' : nombre[0];
43
44      rfc += fechaNacimiento.substr(8, 2);
45      rfc += fechaNacimiento.substr(3, 2);
46      rfc += fechaNacimiento.substr(0, 2);
47
48      return rfc;
49  }
50  };
51

```

Call Stack

#	Function	File:Line
1		

Local Variables

Variable	Value

Display Expressions

Expression	Value
Enter expression to watch	

Breakpoints and Watchpoints

#	Description

21°C
Parc. soleado

Búsqueda

ESP LAA

06:06 p. m.
26/07/2023

OnlineGDB beta
online compiler and debugger for c/c++

code. compile. run. debug. share.

IDE
My Projects
Classroom **new**
Learn Programming
Programming Questions
Sign Up
Login

GOT AN OPINION?
SHARE AND GET REWARDED.
Rakuten AIP

Have fun taking surveys and get paid!

About • FAQ • Blog • Terms of Use • Contact
Us • GDB Tutorial • Credits • Privacy
© 2016 - 2023 GDB Online

SPONSOR Auth0 — Implement Auth0 in any app in just 5 minutes

main.cpp

```

33  for (size_t i = 1; i < apellidoPaterno.length(); i++) {
34      if (apellidoPaterno[i] == 'A' || apellidoPaterno[i] == 'E' ||
35          apellidoPaterno[i] == 'I' || apellidoPaterno[i] == 'O' || apel
36          rfc += apellidoPaterno[i];
37          break;
38      }
39  }
40
41  rfc += (apellidoMaterno.empty()) ? 'X' : apellidoMaterno[0];
42  rfc += (nombre.empty()) ? 'X' : nombre[0];
43
44  rfc += fechaNacimiento.substr(8, 2);
45  rfc += fechaNacimiento.substr(3, 2);
46  rfc += fechaNacimiento.substr(0, 2);
47
48  return rfc;
49  }
50  };
51
52  int main() {
53      Empleado empleado;
54      empleado.CapturarDatos();
55
56      string rfc = empleado.CalcularRFC();
57      cout << "\nEl RFC del empleado es: " << rfc << endl;
58
59      return 0;

```

Call Stack

#	Function	File:Line
1		

Local Variables

Variable	Value

Display Expressions

Expression	Value
Enter expression to watch	

Breakpoints and Watchpoints

#	Description

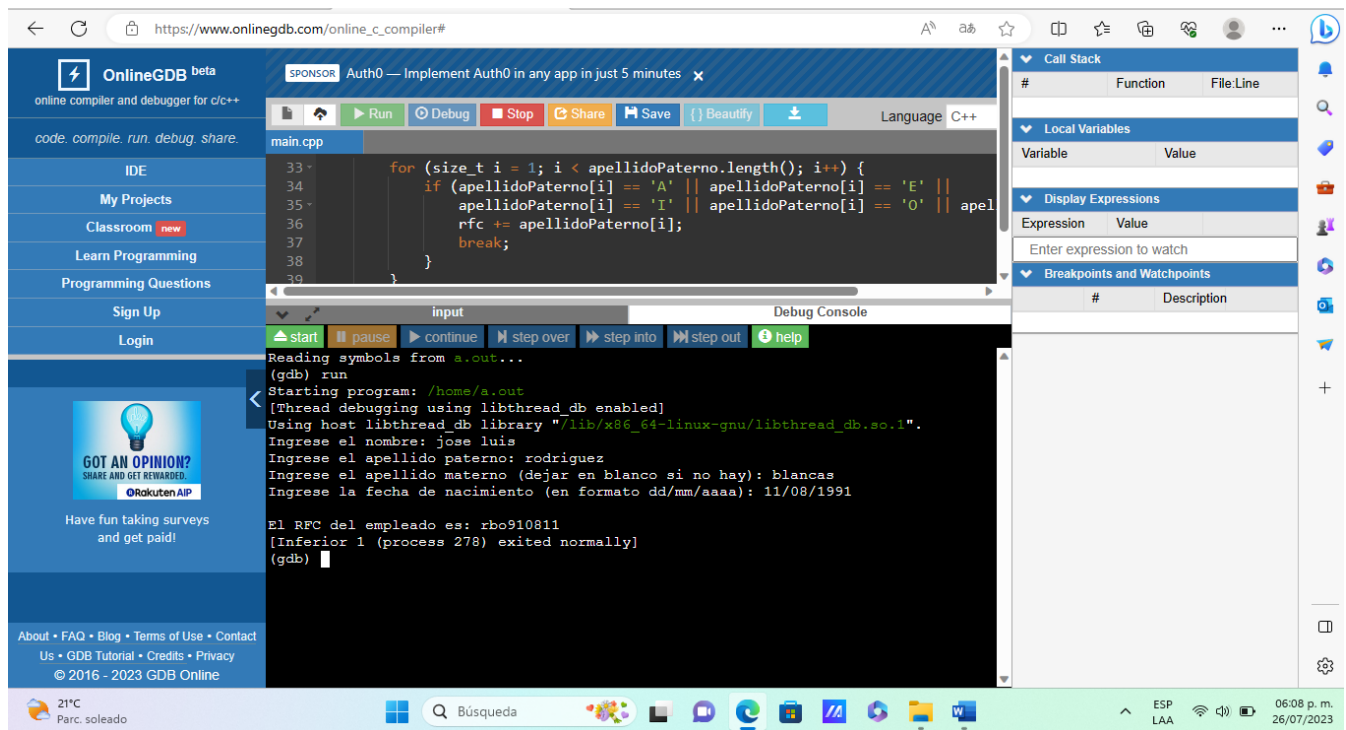
21°C
Parc. soleado

Búsqueda

ESP LAA

06:06 p. m.
26/07/2023

Prueba del sistema:



Este programa utiliza una clase llamada "Empleado" para representar a cada empleado y realizar el cálculo del RFC siguiendo las reglas proporcionadas. La función CapturarDatos() solicita al usuario ingresar los datos del empleado, mientras que la función CalcularRFC() calcula el RFC basándose en los datos capturados.

Una vez que el programa obtiene el RFC, lo muestra en pantalla como resultado final. Cabe mencionar que en esta solución, no se generará la homoclave, ya que es asignada por el SAT y no es parte del proceso que se realiza en esta actividad.

Conclusión

La actividad de crear un programa en lenguaje C++ para calcular el RFC de los nuevos empleados de la constructora AMC ha sido de gran importancia tanto en el campo laboral como en la vida cotidiana. Esta

tarea nos ha permitido aplicar conceptos fundamentales de programación orientada a objetos, como el uso de clases y objetos para estructurar y organizar el código de manera más eficiente.

En el campo laboral, esta actividad tiene un impacto significativo en la automatización de procesos. Al desarrollar esta aplicación, hemos logrado simplificar y agilizar el cálculo del RFC de los empleados, lo que se traduce en un ahorro de tiempo y recursos para la empresa. La precisión y consistencia en la generación del RFC garantiza un cumplimiento adecuado de las obligaciones fiscales y legales, evitando posibles errores en la identificación de los empleados ante el SAT.

En la vida cotidiana, este tipo de habilidades de programación también pueden ser útiles en la resolución de diversos problemas y tareas. La capacidad de crear programas que realicen tareas específicas de forma automática puede ahorrar tiempo en tareas repetitivas y permitirnos ser más eficientes en nuestra vida diaria.

En conclusión, el desarrollo de esta aplicación utilizando conceptos de Objetos y Clases en C++ tiene una relevancia significativa tanto en el ámbito laboral como en nuestra vida diaria. Estas habilidades nos permiten ser más productivos, precisos y eficientes, lo que se traduce en beneficios para las empresas y en una mejor gestión de tareas en nuestra vida personal.

Referencias

1. "The C++ Programming Language" by Bjarne Stroustrup: Este es un libro escrito por el creador de C++, Bjarne Stroustrup. Es una referencia completa sobre el lenguaje y su evolución.
2. "Effective C++" by Scott Meyers: Este libro ofrece una guía detallada sobre cómo escribir código C++ eficiente y seguro.

