

Actividad 1 - Investigación

Lenguajes de programación 1

Ingeniería en Desarrollo de Software

Tutor: Elizabeth Guevara Roa

Alumno: Fernando Pedraza Garate

Fecha: 08 de Julio del 2022

Índice

Etapa 1 – Investigación en lenguaje C++

- Introducción. pág. 3
- Investigación. pág. 4-7
 - ¿Qué es C++?
 - ¿Crees que es muy útil utilizar este lenguaje?, ¿por qué?
 - ¿Qué son los objetos en C++?
 - ¿Qué son las clases en C++?
 - ¿Cómo se utiliza cada uno? Escribe ejemplos
- Conclusión. pág. 8
- Referencias. pág. 9

Introducción

Origen de C++

Fue diseñado a mediados de los años 80 por el danés **Bjarne Stroustrup**. Su intención fue la de extender el lenguaje de programación C para que tuviese los mecanismos necesarios para manipular objetos. Por lo tanto C++ contiene los paradigmas de la programación estructurada y orientada a objetos, por lo que se le conoce como un lenguaje de programación multiparadigma.

En futuras actividades se crearán distintos programas en lenguaje C++, por lo que es necesario tener un amplio conocimiento sobre el mismo para la realización de futuros sistemas informáticos que nos ayudarán a resolver problemas.

Investigación.

¿Qué es C++?

A “C++ “ primero se le conoció como “C con clases”. Luego se cambió a “C++” que significa “incremento de C”, dando a entender que se trata de una extensión del lenguaje de programación C.

¿Crees que es muy útil utilizar este lenguaje?, ¿por qué?

Si, por que en la actualidad las aplicaciones del lenguaje C++ son muy extensas. Podemos nombrar que navegadores WEB, Sistemas operativos, Bases de datos, bibliotecas, aplicaciones gráficas, nubes, videojuegos, compiladores, y más están escritos, o tienen bastante de su estructura, programada en C ++.

¿Qué son los objetos en C++?

Un objeto es una instancia de una clase, es decir una entidad que se construye a partir de las descripciones consignadas en una clase (datos y funciones). Por tanto, un objeto se puede entender como una "variable" que se declara del tipo de dato de cierta clase. Un objeto es como tal la entidad tangible que permite acceder a los datos y funciones modeladas al interior de la clase. Dicho de otro modo, un *objeto de función*, o *functor*, es cualquier tipo que implementa

operator()). Este operador se conoce como el *operador de llamada* o a veces el *operador de la aplicación*. La biblioteca estándar de C++ usa objetos de función principalmente como criterios de ordenación para los contenedores y en algoritmos.

Los objetos de función proporcionan dos ventajas principales en comparación con una llamada de función sencilla. La primera es que un objeto de función puede contener el estado. La segunda es que un objeto de función es un tipo y, por tanto, puede usarse como un parámetro de plantilla.

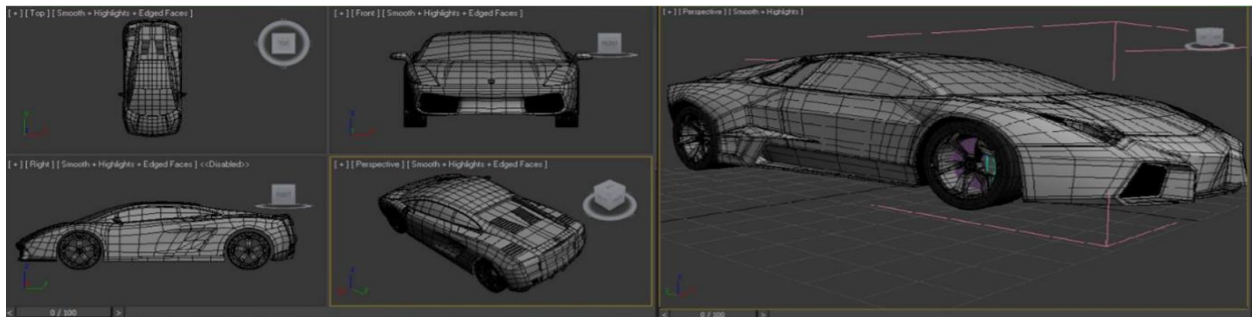
¿Qué son las clases en C++?

Una clase es en general un modelo, receta o plantilla que define el estado y comportamiento de cierto tipo de objetos. Una clase puede pensarse como una colección de variables (atributos o propiedades) y funciones (métodos) que permiten representar un conjunto de datos y especificar las operaciones o procedimientos que permiten manipular tales datos. Se puede inclusive entender una clase como un tipo de dato personalizado, similar a las estructuras (structs), donde cada programador define los miembros que va a tener su tipo de dato. De hecho, los tipos de dato nativos de C++ son en realidad clases.

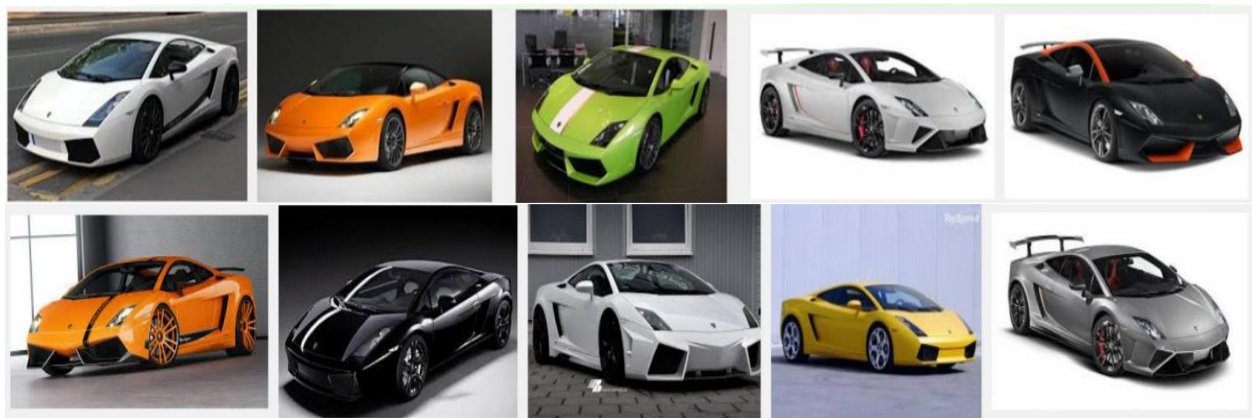
¿Cómo se utiliza cada uno? Escribe ejemplos

Una analogía para entender las clases y los objetos puede ser una fábrica ensambladora de carros. Hay un modelo o diseño (clase) específico de un auto, pero este modelo en si no es un carro, es solo una descripción de que características y funcionalidades deben tener los carros que sean de ese modelo. Los carros ensamblados en la fábrica de acuerdo a dicho modelo serían los objetos, es decir entidades tangibles que se construyeron a partir de las descripciones y especificaciones consignadas en el diseño o modelo (o sea la clase).

Clase:



Objetos:



Para declarar una clase en C++ se utiliza la palabra reservada `class`, se da un nombre a la clase y luego entre llaves se declaran los miembros de la clase. Las clases no pueden declararse al interior de funciones, ya que son una definición de un tipo de dato creado por el usuario (programador). En general, las clases se declaran en bibliotecas (librerías) individuales cuyo nombre es usualmente el mismo nombre de la clase.

```
class MiClase
{
    //Aquí van los miembros de la clase: Variables y funciones
}; //NO olvidar el ;
```

Los objetos, tal como se había mencionado con anterioridad, son variables (instancias) del tipo de dato definido por una clase. Por tanto, los objetos se pueden declarar al interior o por fuera de funciones, tal y como una variable local o global respectivamente. Pueden ser declarados como miembros de otras clases, es decir al interior de otras clases. Luego, para declarar un objeto primero se utiliza el nombre de la clase a la que pertenece el objeto seguido de un nombre para el objeto y de una lista opcional de inicialización entre paréntesis. Dicha lista se verá más adelante.

```
MiClase objetoGlobal; //Declaración de un objeto global de la clase MiClase
```

```
int main()
{
    MiClase objetoLocal; //Declaración de un objeto local de la clase
    MiClase
}
```

Conclusión.

En conclusión es importante comprender la base del lenguaje de C++ para poder desarrollar y elaborar un buen programa, reconociendo la diferencia entre las clases y los objetos, así como su estructura, para no cometer tener errores al momento de ejecutar los programas, respetando en todo momento las llaves, corchetes, y el punto y coma, ya que son componentes de esta misma.

En cuestión de utilidad es muy diverso al tener oportunidad de aplicarse en diferentes tipos de desarrollo, como ya se mencionó, páginas WEB, video juegos, BD, nubes, etc.

file:///C:/Users/ferna/Documents/Universdidad%20IDS/Segundo%20bloque/Lenguajes%20de%20programaci%C3%B3n%201/COP_L_LP1_TU.pdf

[https://docs.microsoft.com/es-es/cpp/standard-library/function-objects-in-the-stl?view=msvc-](https://docs.microsoft.com/es-es/cpp/standard-library/function-objects-in-the-stl?view=msvc-170)

[c#:~:text=Una%20clase%20puede%20pensarse%20como,que%20permiten%20manipular%20tales%20datos.](#)

c#:~:text=Para%20declarar%20una%20clase%20en,por%20el%20usuario%20(programador).