|  |  |  |
| --- | --- | --- |
| Universidad Centroamericana “José Simeón Cañas” | |  |
| Facultad de Ingeniería y Arquitectura | |
|  | |
|  | |
| Tutorías - Fundamentos de Programación | Ciclo 01/2021 |
| Tutor: Ernesto José Canales Guillén | |

**Introducción a la programación en C++**

**Comentarios**

El programa que escriba debe ser claro no solo para usted, sino también para el lector de su programa. Parte de una buena programación es la inclusión de comentarios en el programa. Normalmente, los comentarios se pueden utilizar para identificar a los autores del programa, dar la fecha cuando el programa esté escrito o modificado, dé una breve explicación del programa, y explicar el significado de las declaraciones clave en un programa. Los comentarios son para el lector, no para el compilador. Entonces, cuando un compilador compila un programa para comprobar los errores de sintaxis, ignora por completo los comentarios.

Hay dos tipos comunes de comentarios en un programa C++: **comentarios de una sola línea** y **comentarios de varias líneas**.

Los comentarios de una sola línea comienzan con ***//*** y se pueden colocar en cualquier lugar de la línea. Todo encontrado en esa línea después de ***//*** es ignorado por el compilador.

Los comentarios de varias líneas se incluyen entre ***/\**** y ***\*/***. El compilador ignora cualquier cosa que aparece entre ***/\**** y ***\*/***.

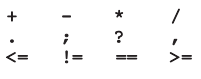


**Token**

La unidad individual más pequeña de un programa escrito en cualquier lenguaje se llama token. Los tokens de C++ se dividen en **símbolos especiales**, **símbolos de palabras** e **identificadores**.

**Símbolos especiales (Ejemplos)**

La primera fila incluye símbolos matemáticos para sumar, restar, multiplicar y división. La segunda fila consta de signos de puntuación tomados de la gramática inglesa. La tercera fila consta de tokens compuestos por dos caracteres que se consideran un solo símbolo. Ningún carácter puede interponerse entre los dos caracteres de la ficha, ni siquiera un espacio en blanco.



**Identificadores**

Los identificadores constan de letras, dígitos y el carácter de subrayado (\_) y deben comenzar con una letra o un guion bajo; son nombres de cosas que aparecen en programas, como variables, constantes y funciones. Todos los identificadores deben obedecer a C ++ reglas para identificadores. A diferencia de las palabras reservadas, se pueden redefinir identificadores predefinidos, pero no sería prudente hacerlo.

C++ distingue entre mayúsculas y minúsculas: las letras mayúsculas y minúsculas se consideran diferentes. Por lo tanto, el identificador *NUMERO* no es el mismo que el identificador *numero*. Del mismo modo, los identificadores *X* y *x* son diferentes.

**Palabras reservadas (Keywords)**

Las palabras reservadas también se denominan palabras clave. Las letras que componen una palabra reservada son siempre en minúsculas. Al igual que los símbolos especiales, cada uno se considera un símbolo único. Además, los símbolos de palabras no se pueden redefinir dentro de ningún programa; es decir, no pueden utilizarse para cualquier otro uso que no sea el previsto.



**Espacios en blanco (Whitespaces)**

Cada programa de C ++ contiene espacios en blanco. Los espacios en blanco incluyen espacios en blanco, tabulaciones y nueva línea de caracteres. En un programa de C ++, los espacios en blanco se utilizan para separar símbolos especiales, reservados palabras e identificadores. Los espacios en blanco no se pueden imprimir en el sentido de que cuando se impreso en una hoja de papel blanco, el espacio entre símbolos especiales, palabras reservadas e identificadores es blanco. La utilización adecuada de espacios en blanco en un programa es importante. Ellos pueden ser utilizado para hacer que el programa sea más legible.

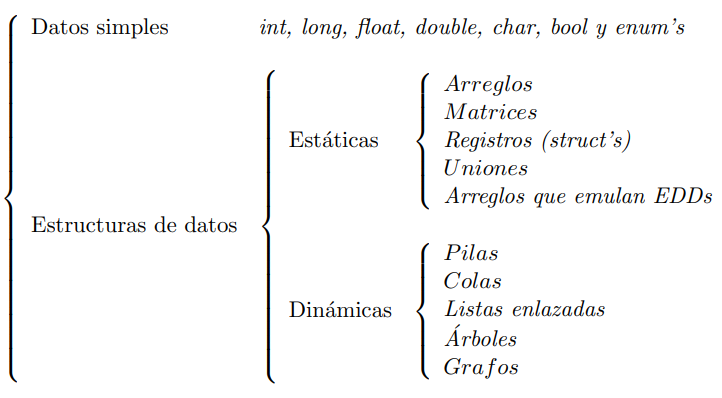
**Tipos de datos**

El objetivo de un programa en C++ es manipular datos. Diferentes programas manipulan datos diferentes. Un programa diseñado para calcular el cheque de pago de un empleado sumará, restará, multiplicar y dividir números, y algunos de los números pueden representar horas trabajadas y tarifa de pago. De manera similar, un programa diseñado para alfabetizar una lista de clases manipulará los nombres. Tú no usarías una receta de pastel de cerezas para a hornear galletas.

Del mismo modo, no usarías un programa diseñado para realizar cálculos aritméticos para manipular caracteres alfabéticos. Además, no multiplicaría ni restaría nombres. Reflejando este tipo de subyacentes diferencias, C++ categoriza los datos en diferentes tipos, y solo ciertas operaciones pueden ser realizado en tipos particulares de datos. Aunque al principio pueda parecer confuso, por ser tan tipo consciente, C++ tiene controles incorporados para protegerse contra errores.

C++ tiene tres categorías:

* Tipos de datos simples.
* Tipos de datos estructurados.
* Punteros.



**Tipos de datos simples**

El tipo de datos simple es el tipo de datos fundamental en C++ porque se convierte en un bloque base para el tipo de datos estructurados.

Hay ciertas categorías de datos simples:

* **Enteros**, que es un tipo de datos que trata con números enteros o números sin una parte decimal.
* **Punto flotante**, que es un tipo de datos que trata con números decimales.
* **Caracteres**, que son un tipo dato que trata con letras, dígitos, símbolos y signos de puntuación.
* **Booleanos**, que son un tipo de dato que trata con 0s y 1s, verdadero y falso.
* **Enumeraciones**, que es un tipo de datos definido por el usuario.

Datos base:



Datos base con modificadores:

