C++ Programming Real Time Programming



◆ OBJECT & CLASS

◆ <u>Comments</u>







OBJECT & CLASS

- The world is full of objects.
 Objects have state and behavior
 State the features that describes it (properties):
 engine type, make, model, color, etc.
 Behavior the things that it knows how to do
 (e.g.) accelerate, brake, change gear, etc.
- An object can be: A physical thing in the real world A representation of reality A tangible or visible things A thing to which action or though can be directed Passive do nothing until activated (e.g. switch) Active continually monitoring until conditions change (e.g. thermostat)
- Class- An object is an instance or example of category class
 Classes can use as a template for creating objects Example:
 Houses apartment, bungalow
 People– brothers class, mothers class, students class.
- Ex- Create at least 3 new classes for each real-world object that you know. Defines at least 3 operations and 3 attributes of each class.
 For each new class that you have created above, create at least 3 objects

Comments

When you are writing a program: • always clear • selfevident what you are trying to do. • Comments are simply text that is ignored by the compiler, but that may inform the reader of what you are doing at any particular point in your program. • C++ comments come in two flavors: • the double-slash(//) comment, • the slash-star (/*) comment.

<u>variable</u>

- Data Types int This data type is used to represent short integer.
 int This data type is used to represent integer.
 long int This data type is used to represent long integer.
 float This data type is used to represent long integer.
- double This data type is used to represent double precision floating point number.
 long double - This data type is used to represent double precision floating point number.
- char This data type is used to represent a single character.
 bool -This data type is used to represent boolean value.

Variables

A variable is the content of a memory location that stores a certain value. • A variable is identified or denoted by a variable name. • The syntax for declaring variable name: • data_typevariable_name; • int number; type identifier.

Declaring Variable

int a; • This declares a variable name a of type int. • If there exists more than one variable of the same type, such variables can be represented by separating variable names using comma. • For instance: • int x,y,z; • This declares 3 variables x, y and z all of data type int.



Like variables, constants are data storage locations. • Unlike variables, and as the name implies, constants don't change. • You must initialize a constant when you create it, and you cannot assign a new value later. • C++ has two types of constants: literal and symbolic.

Literal Constants

A literal constant is a value typed directly into your program wherever it is needed. • For example • intmyAge = 39; • myAge is a variable of type int; • 39 is a literal constant. • You can't assign a value to 39, and its value can't be changed.

 \diamond

Integer Data Types: Valid Ranges

- ♦ u8 : 0 255
- ♦ s8 : -128 127
- ♦ u16 : 0 65535
- ♦ s16 : -32768 32767
- ♦ u32 : 0 4.294967295e9
- ♦ s32 : -2.147483648e9 2.147483647e9

Note: ranges given are decimal.

Pointer Arithmetic

- Pointers store memory addresses just numbers telling the processor which memory cell to access.
- Adding 1 to a pointer makes it point to the next memory location.
- Subtracting 1 from a pointer makes it point to the previous memory location
- Subtracting two pointers from each other shows how much space is between the memory locations pointed to by the pointers.
- Pointers "know" the sizes of the variables they point to.
- adding to an int pointer will probably result in a different address than adding to a char pointer