No I/O is built into C++

- instead, a library provides input stream and output stream

\[\text{Keyboard} \xrightarrow{\text{istream}} \text{executing program} \xrightarrow{\text{ostream}} \text{Screen}\]

<iostream> is header file

- for a library that defines 3 objects
  - an istream object named cin (keyboard)
  - an ostream object named cout (screen)
  - an ostream object named cerr (screen)
**Insertion Operator ( << )**

- the insertion operator `<<` takes 2 operands
- the left operand is a stream expression, such as `cout`
- the right operand is an expression of simple type, or a string, or a manipulator

**Output Statements**

SYNTAX (revised)

```cpp
cout << ExpressionOrManipulator
    << ExpressionOrManipulator ...;
```
Output Statements

SYNTAX

\[ \text{cout} \ll \text{Expression} \ll \text{Expression} \ldots; \]

These examples yield the same output.

\[ \text{cout} \ll \text{“The answer is “;} \]
\[ \text{cout} \ll 3 \* 4; \]
\[ \text{cout} \ll \text{“The answer is “} \ll 3 \* 4; \]

Giving a Value to a Variable

In your program you can assign (give) a value to the variable by using the assignment operator =

\[ \text{ageOfDog} = 12; \]

or by another method, such as

\[ \text{cout} \ll \text{“How old is your dog?”;} \]
\[ \text{cin} \gg \text{ageOfDog}; \]
>> is a binary operator

>> is called the input or extraction operator

>> is left associative

<table>
<thead>
<tr>
<th>EXPRESSION</th>
<th>HAS VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>cin  &gt;&gt;  age</td>
<td>cin</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>STATEMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>cin  &gt;&gt;  age  &gt;&gt;  weight ;</td>
</tr>
</tbody>
</table>

Extraction Operator ( >> )

- variable cin is predefined to denote an input stream from the standard input device (the keyboard)

- the extraction operator >> called “get from” takes 2 operands. The left operand is a stream expression, such as cin—the right operand is a variable of simple type.

- operator >> attempts to extract the next item from the input stream and store its value in the right operand variable
Input Statements

SYNTAX

```cpp
cin >> Variable >> Variable . . . ;
```

These examples yield the same result.

```cpp
cin >> length ;
cin >> width ;
cin >> length >> width ;
```

Whitespace Characters Include . . .

- blanks
- tabs
- end-of-line (newline) characters

The newline character is created by hitting Enter or Return at the keyboard, or by using the manipulator endl or “\n” in a program.