

Quick Intro to C++

- **Statements**
 - Sequence
 - Decision
 - Loop
- **Data Types**
 - Simple
 - Compound
- **I/O Statements**
- **Functions**

C++00b

C++ Statements

▶ **Sequence:**

- **Assignment**

```
a = b;  
c = 3 + 2 * d;  
r = 5 * (s = t); // Legal?
```

- **Compound Statement: { }**

Used to put several statements where normally only one statement goes.

C++002b

C++ Statements

► Decision:

- if Statement:

```
if (expr) S  
if (expr) S else S
```

- switch - case Statement:

```
switch(expr) {  
    case val: S  
             ⋮  
             S  
            break;  
    case val: S  
             ⋮  
             S  
            break;  
    default: S  
            ⋮  
            S  
            break;  
}
```

C++0030

C++ Statements

► Loop:

- while Statement:

```
while(expr) S
```

- do-while:

```
do S while(expr);
```

- for:

```
for(S1; expr1; expr2) S2
```

Equivalent to:

```
S1  
while(expr1) {  
    S2  
    expr2;  
}
```

C++0040

C++ Data Types

▶ **Simple Data Types:** ▶ **Compound Data Types:**

```
int
long int
short int
unsigned int
char
float
double
pointer (&)
void
```

```
Array
struct
class
```

▶ **User Defined:**

```
enum
typedef
```

C++0050

C++ Input/Output

▶ **Standard Input/Output:**

```
#include<iostream.h>
cin
cout
```

▶ **Files (streams):**

```
#include<fstream.h>
```

```
ifstream strname;
ofstream strname;
```

```
strname.open(fname);
strname.close();
strname.fail();
```

C++0060

C++ Functions

- ▶ Prototype – must appear before any use of function

Examples:

```
int func1(int, int, float);
```

NOTE semicolon!
argument types (names not required)

function name
type that function will return

```
void func2(int, int, float&);
```

reference argument!
returns NO value in function name!!

```
char func3(char, int[]);
```

array (always passed by reference!)

C++0070

C++ Functions

- ▶ Function definition (actual code for function)

Example:

```
int func1(int a, int b, float x)
{
    // local declarations
    float c;

    // code
    c = float(a)/float(b) + x;
    return(c);
} // end of func1
```

type casts

C++0080

C++ Functions

► **Function definition (actual code for function)**

Example 2:

```
void func2(int a, int b, float& x)
{
    // local declarations
    int r, s;

    // code
    r = a + b;
    s = a - b;
    x = float(r + s);
} // end of func2
```

value returned in argument list (NO return!)

C++0090