

CSC230

Intro to C++

C++ Overview

2

- Middle-level language
- Superset of C
- Statically typed (compile-time check, not run-time)
- System and Application programming
- Case sensitive

C++ Overview

3

- Object-oriented

- ▣ Encapsulation

- ▣ Data Hiding

- ▣ Inheritance

- ▣ Polymorphism

- Ansi Standard

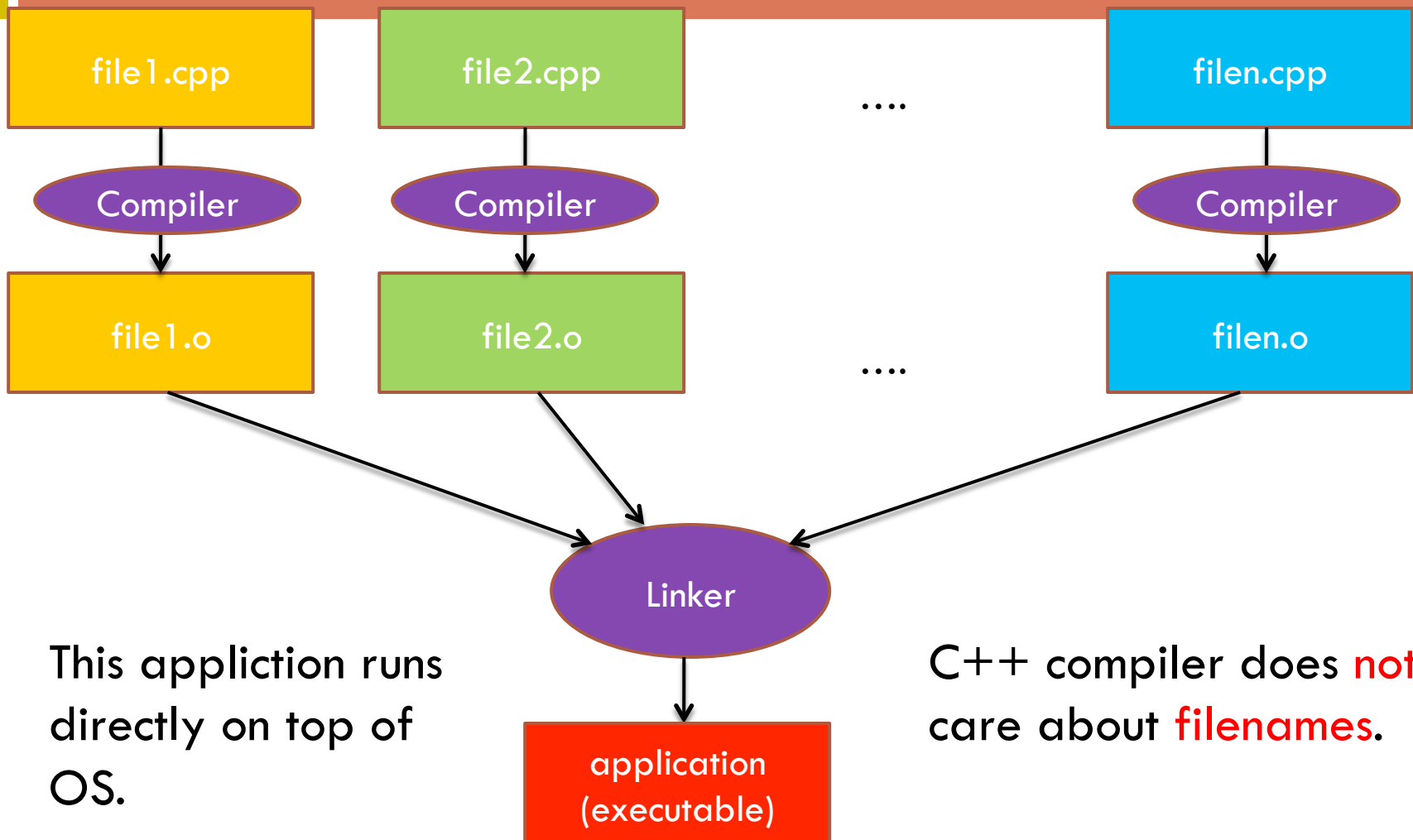
- ▣ Mac, Unix, Windows all support

Why C++?

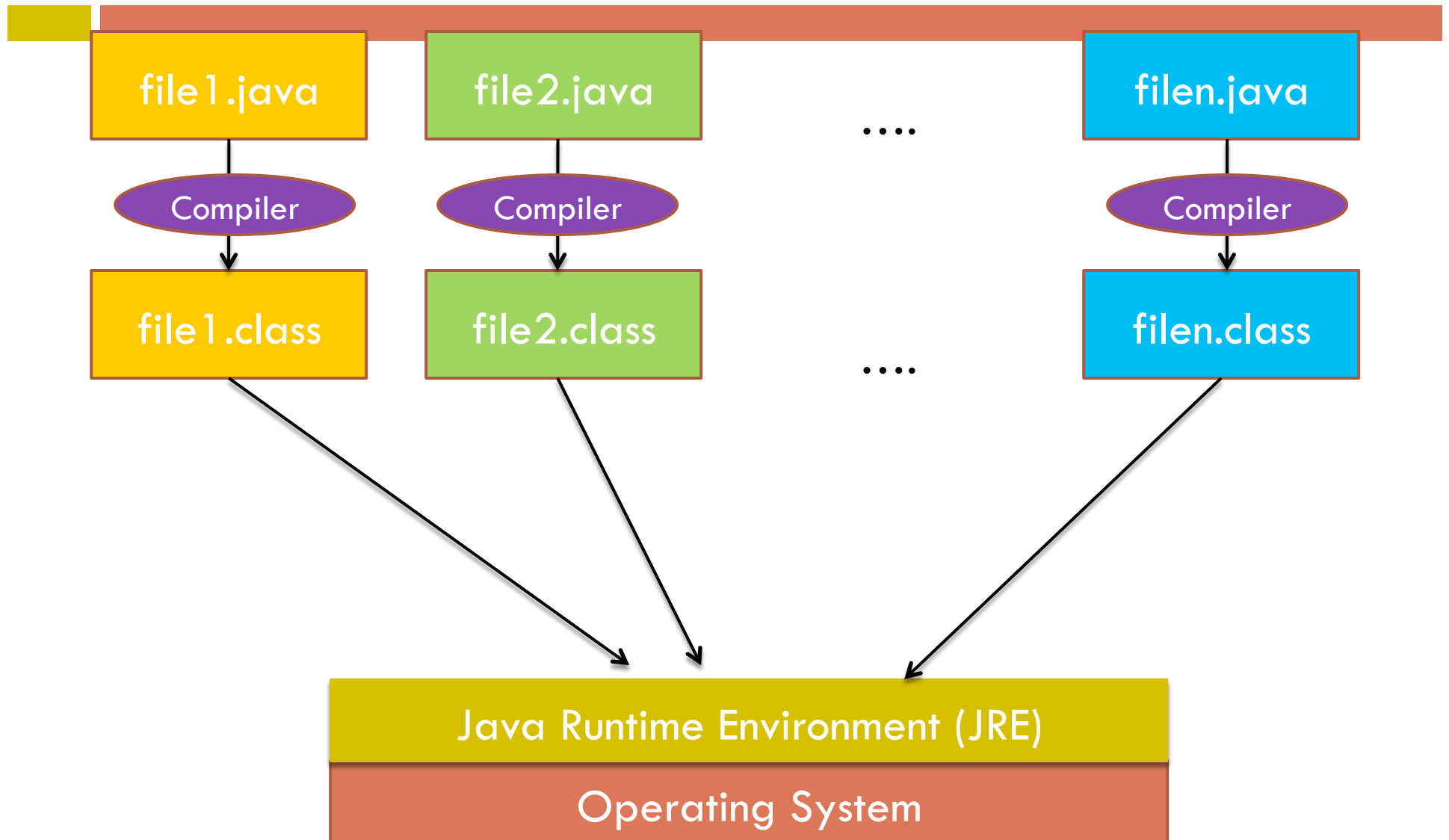
4

- Super **performance** with Object-oriented abstractions
- Control **hardware** (hard drive, camera)
- **Memory** control and **timing** (game design, car/airplane control)
- Examples

C++ compiler & Linker usage



Java compiler usage



C++ vs. Java

7

	C++	Java
Focus	Efficiency	Productivity
Memory Management	Arbitrary memory access	Through object, automatic garbage collection
Code	concise	explicit
Type Casting	Restricted	Only for compatible types
Programming Paradigm	Procedural, or Object-oriented	Object-Oriented
Preprocess	Yes	No
Main Advantage	Powerful capabilities	Feature-rich, easy use
Input	<code>cin >> x;</code>	<code>x = scan.nextInt();</code>
Output	<code>cout << y;</code>	<code>System.out.println(y);</code>

C++ vs. Java

8



Pilot

Ready to fly?



Driver

How would you go to grocery store,
fly or drive?

Install C++?

9

- Linux/Unix

- ▣ <http://gcc.gnu.org/install/>

- Mac OS X

- ▣ Xcode

- Windows

- ▣ MinWG, www.mingw.org

- ▣ gcc-core, gcc-g++, binutils, MinGW runtime

- ▣ Change PATH environment variable

“Hello, World” in C++

10

```
#include <iostream>           ← Header File
using namespace std;          ← Namespace

int main()                     ← Main function
{
    cout << "Hello World";    ← output
    return 0;                 ← return
}
```

Compile & Execute C++ code

11

- Text Editor
- Save file as: HiWorld.cpp
- Open a terminal
- `g++ HiWorld.cpp`
- `./a.out`

C++ Identifiers

12

- A-Z, a-z or _ followed by letters, _ or 0-9
- No punctuation characters, @, \$, %
- Case sensitive Hello ≠ hello

C++ Comments

13

How to make comments in C++?

`/* C++ comments can`

`* span multiple lines`

`* /`

`// Comment starts from here until the end of the line`

C++ Data Types

14

- **Primitive**

- ▣ `bool, char, int, float, double, void`

- **Modifier**

- ▣ `signed, unsigned, short, long`

Size Matters

15

Type	Size
bool, char, unsigned char, signed char	1 byte
short, unsigned short, wchar_t	2 byte
float, int, unsigned int, long, unsigned long	4 byte
double, long double, long long	8 byte

Modifier + Primitive?

16

Type	Width	Range
char	1 byte	-127 to 127 0 to 255
unsigned char	1 byte	0 to 255
signed char	1 byte	Same as char
int	4 bytes	-2147483648 to 2147483647
unsigned int	4 bytes	0 to 4294967295
signed int	4 bytes	Same as int
short int	2 bytes	-32768 to 32767

Type	Width	Range
unsigned short int	2 bytes	0 to 65535
signed short int	2 byte	Same as “short int”
long int	4 bytes	Same as int
signed long int	4 bytes	Same as long int
unsigned long int	4 bytes	Same as “unsigned int”
float	4 bytes	+/- 3.4e +/-38
double	8 bytes	+/- 1.7e +/-308
long double	8 bytes	Same as double

Check the size of some type?

17

```
#include <iostream>
using namespace std;
```

```
int main()
{
    cout << "Size of float : " << sizeof(float) << endl;
    cout << "Size of double : " << sizeof(double) << endl;
    return 0;
}
```

What if no “endl” ?

<<  Put-to

>>  Get-from

Do not like the type name?

18

- No problem, redefine the type name

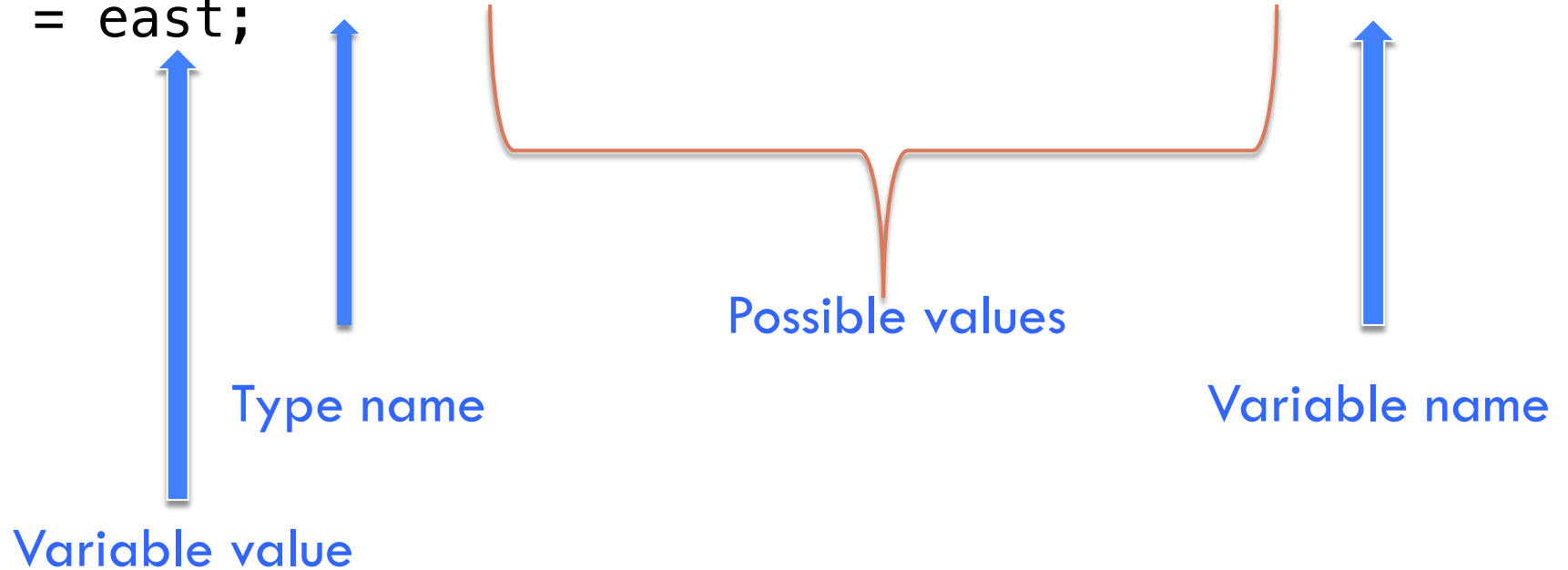
```
typedef double weight;  
weight cuurentWeight;
```

```
double  weight
```

Enumerated Types

19

```
enum direction {east, west, south, north } d;  
d = east;
```



How to define/use a function?

20

```
#include <iostream>
using namespace std;

int func(){
    return 200;
}

int main(){
    cout << func() << endl;
}
```

```
#include <iostream>
using namespace std;

int func();

int main(){
    cout << func() << endl;
}

int func(){
    return 200;
}
```

Before calling a function, **define** it **OR** **declare** it.

Use a function from a different file?

21

```
#include <iostream>
using namespace std;
extern int a;
extern int func();
int main()
{
    cout << "a value : " << a << endl;
    cout << "func value : " << func() << endl;
    return 0;
}
```

↑
one.cpp

```
#include <iostream>
using namespace std;

int a = 100;

int func(){
    return 200;
}
```

↑
two.cpp

g++ one.cpp two.cpp

Local/Global Variables

22

```
#include <iostream>
using namespace std;
```

```
int a = 100;
```

← global

```
int main()
{
```

```
    int b = a;
```

```
    for(int c = 0; c < 10; c++)
```

```
        cout << c << endl;
```

```
    return 0;
```

```
}
```

← local

Examples: local / global

Local/Global Variables

23

□ Local variables

- ▣ Defined **inside** a function or block
- ▣ Used **inside** the same function or block

□ Global variables

- ▣ Defined **outside** any function
- ▣ Used by **all** functions



11,000 global variables in Toyota cars!!!!
Are you kidding me ?

<http://www.i-programmer.info/news/91-hardware/6995-toyota-code-could-be-lethal-.html>

Namespace

24

- Namespace defines a **scope**.
- The names within the **same scope** can not be the same
- The names from two **different** namespaces (**scopes**) can be the same.

Examples: namespaces

```
#include <iostream>
using namespace std;

namespace first_space{
    void func(){
        cout << "Inside first_space" << endl;
    }
}

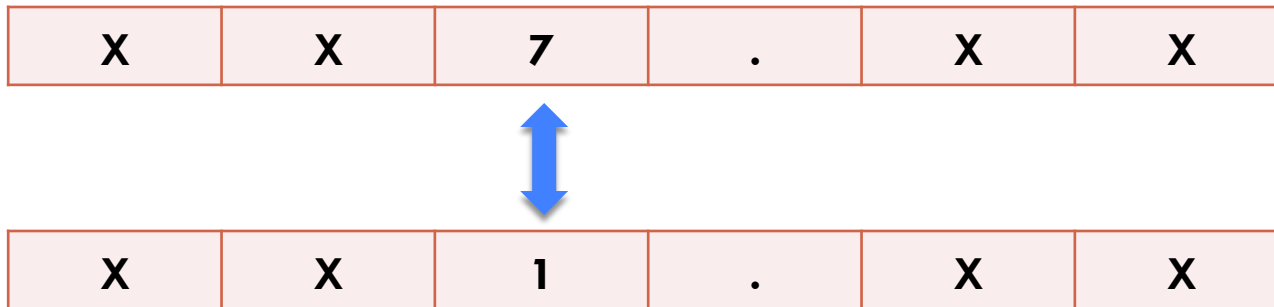
namespace second_space{
    void func(){
        cout << "Inside second_space" << endl;
    }
}

int main ()
{
    first_space::func();
    second_space::func();
    return 0;
}
```

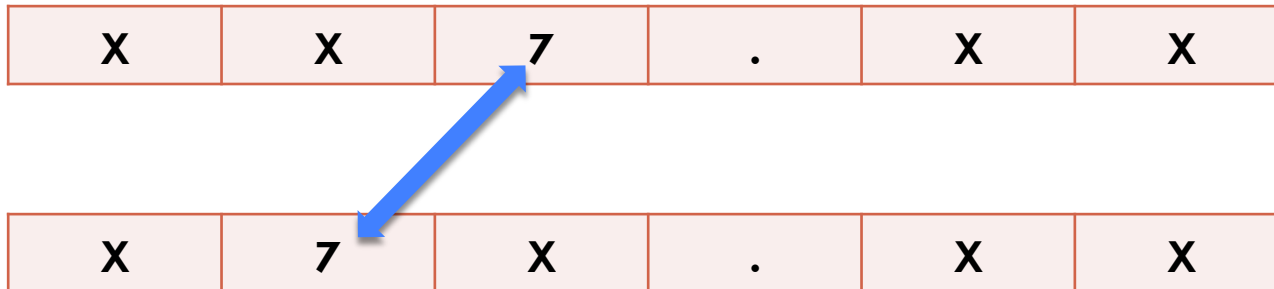

What does a digit mean?

25

Value matters



Position matters



What does a digit mean?

26

Base matters

Q: How tall are you?

90 inches



Count by 10

7.5 feet



Count by 12

base

Same height, different representations

Numerical bases

27

Base	Symbols	Do you use it
10	0~9	Yes
2	0, 1	No
8	0~7	No
16	0~9, A~F	No

What does a sequence of digits mean?

28

	2	1	0		-1	-2
Decimal	1	8	7	.	2	6

$$1 \times 10^2 + 8 \times 10^1 + 7 \times 10^0 + 2 \times 10^{-1} + 6 \times 10^{-2} = 187.26$$

	2	1	0		-1	-2
Binary	1	0	1	.	1	1

$$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} = 5.75$$

Octal & Hexdecimal

29

	4	3	2	1	0
Octal	1	4	7	2	6

$$1 \times 8^4 + 4 \times 8^3 + 7 \times 8^2 + 2 \times 8^1 + 6 \times 8^0 = 6614$$

	4	3	2	1	0
Hex	1	A	7	C	6

$$1 \times 16^4 + 10 \times 16^3 + 7 \times 16^2 + 12 \times 16^1 + 6 \times 16^0 = 108486$$

Hexadecimal	Decimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10
B	11
C	12
D	13
E	14
F	15

Use binary, decimal, Octal, hex in C++

31

```
#include <iostream>
using namespace std;
```

```
int main()
{
    int a, b, c, d;
    a = 0b11010;
    b = 0x3A5B78;
    c = 023235;
    d = 12345;
    cout << a << '\t' << b << '\t' << c << '\t' << d << '\n' ;
    return 0;
}
```

0b precedes **binary**
0x precedes **hexadecimal**
0 precedes **octal**
No preceding for **Decimal**

\t: tab key
\n: newline
Other special characters:
****: \ character
\': ' character
\": " character
\r: carriage return

Constant values

32

```
#include <iostream>
using namespace std;
```

```
#define PI 3.14159
```

```
int main()
{
```

```
    const double DAYS = 365;
    cout << PI << '\n' << DAYS << '\n';
    return 0;
```

```
}
```



Both PI and DAYS
are constant

Examples: namespaces