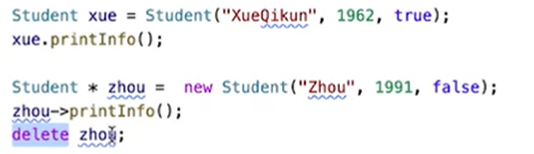
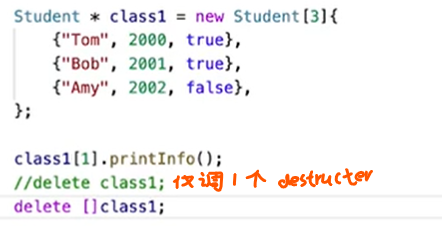
**Dynamic memory**





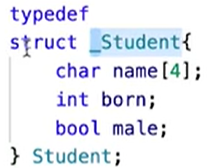
**Compound**

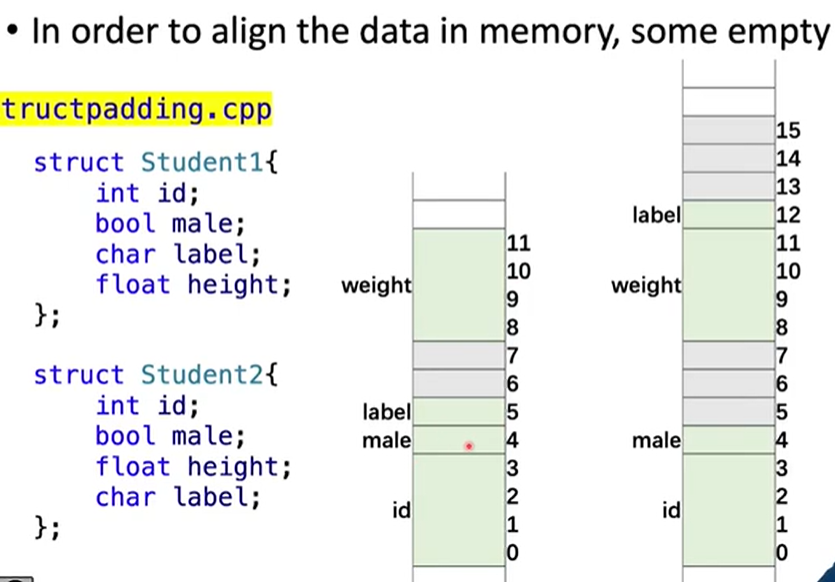
**array, vector** sizeof: size in bytes

**string**   end with \0

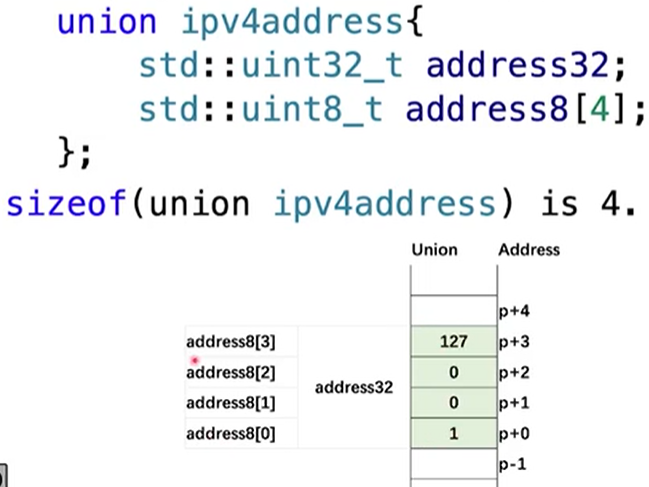
     #include<string> [lec4][lab4]

**struct**

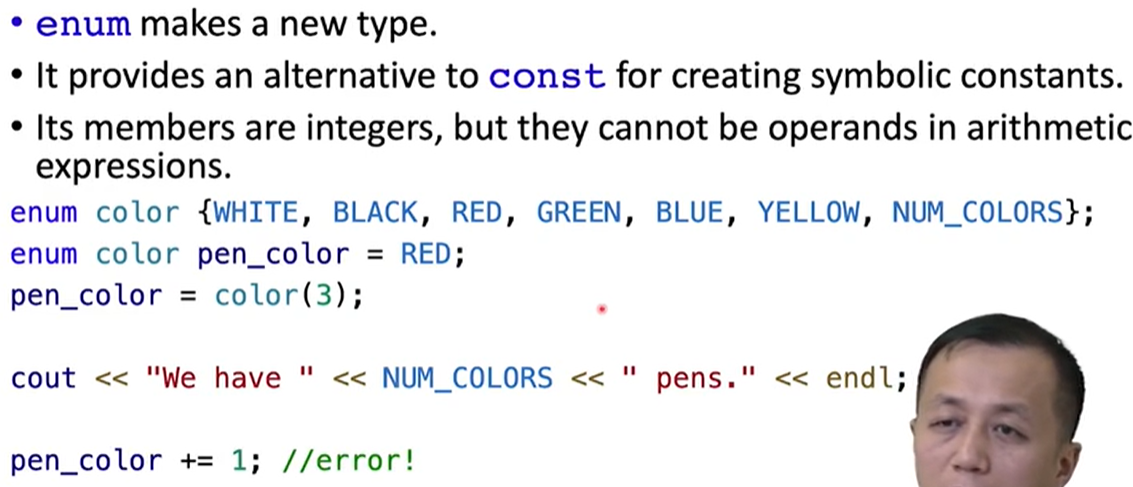
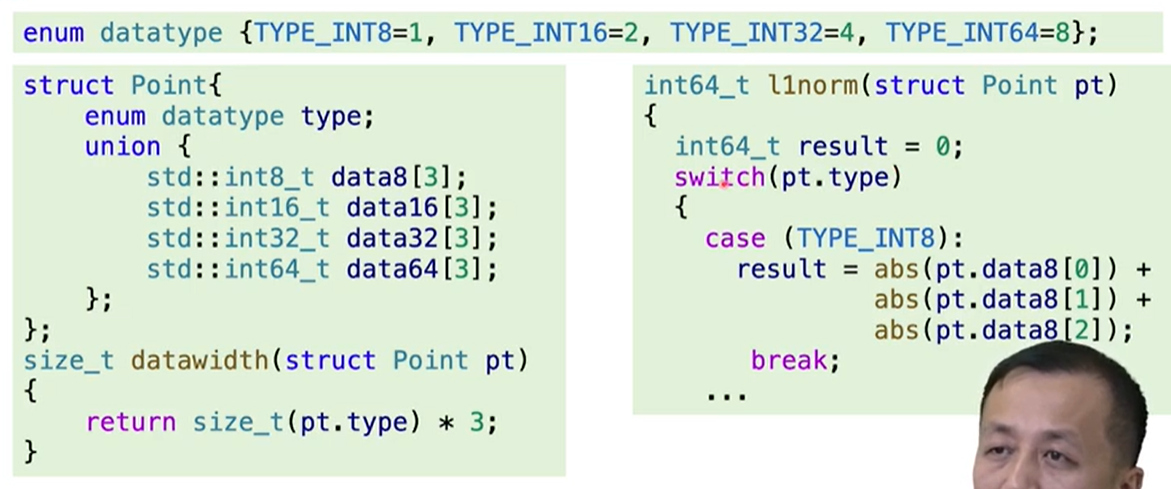
#typedef + alias no need “struct Student s”, Student s



**union**



**enumeration**[lec5p38][lec10p23] z

**Variable Types**

在名称中只能使用字母字符、数字和下划线（\_）。

名称的第一个字符不能是数字。

区分大写字符与小写字符。

不能将C++关键字用作名称。

以两个下划线或下划线和大写字母打头的名称被保留给实现（编译器及其使用的资源）使用。以一个下划线开头的名称被保留给实现，用作全局标识符。

**Integer**

short >=16

int >=short(32)

long >=32 && >=int(32)

longlong >=64 && >=long(64)

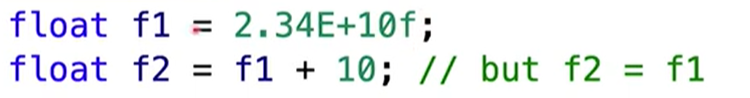
unsigned (31bit is not sign

size\_t[type of sizeOf’s return type]32/64bits

<cstdint>intx\_t uintx\_t INTx\_MIN

**floating**

float  32    7digits



double 64   15digits

long double 64     15digits

**bool**  8 nonzero->true   zero->false

**char** 8 signed/unsigned

**Pointer**

int a;

int \* const p= a; can’t change the address p to

int const \* p1=a; can’t change a through p1/p2

const int \*p2=a;

smart pointer

**reference** especially for a class object

**Class**

C++ automatically provides the following member functions

⮚ A default constructor if you define no constructors ⮚ A copy constructor if you don’t define one ⮚ An assignment operator if you don’t define one ⮚ A default destructor if you don’t define one ⮚ An address operator

Abstraction component:  public interface

Encapsulation component: gather the implementation details

Variable const member variable--[lec10 p23] static member variable--or outside::

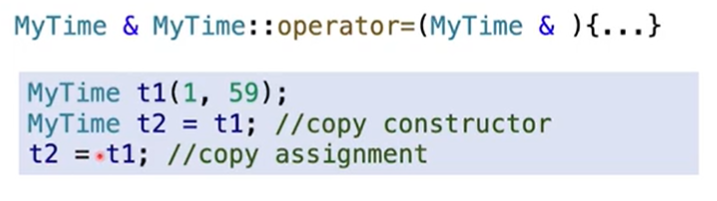
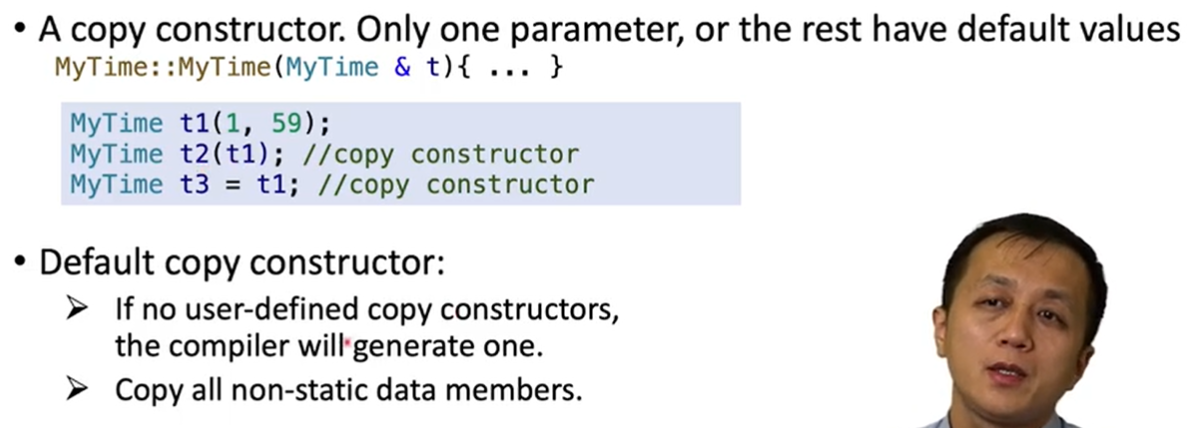
public protected[members and friends of itself&derived class father.n++× this.n++√] private

Function const member functions--A function promises NOT to modify the invoking object

static member functions-- The only data members it can use are the static data members; public:can be invoked using the class name and the scope-resolution operator

Constructor

Copy: operater= | Copy constructor[lec12 p13]



        Type cast: Convertion constructor[lec11 p25] | Conversion function: explicit operater [typename]()[p26]

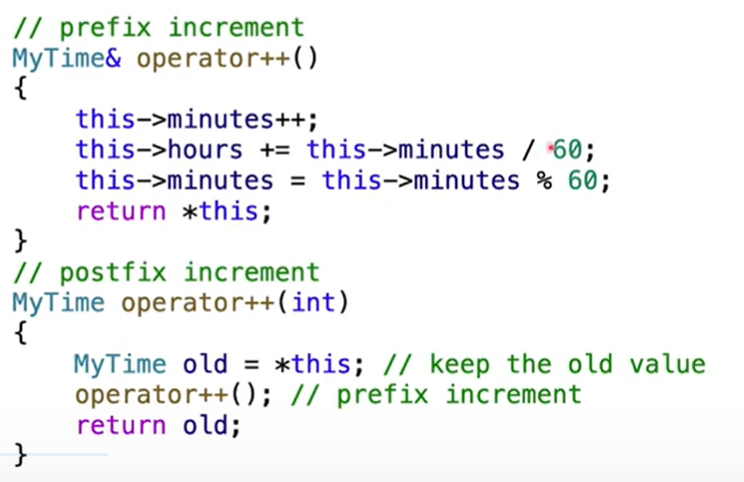
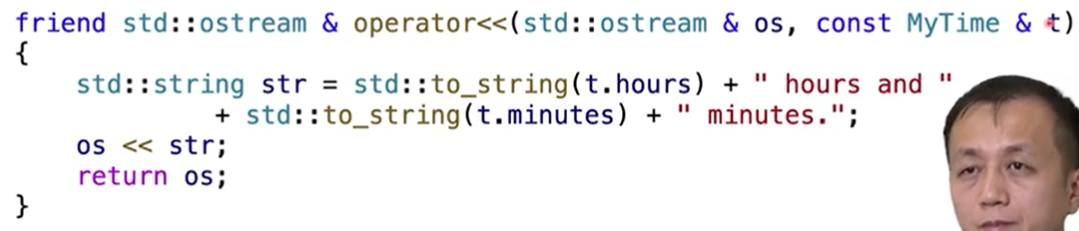
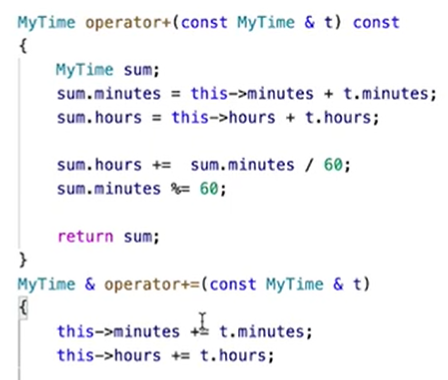
Destructor

        Passing an object as a function argument somehow causes the destructor to be called

Operator overloading

        Use only member functions to overload =,(),[],-> | cannot overload . .\* ?::

Member/nonmenber: Time operator+(const Time &t)const/friend Time operator+( const Time &t1,t2)[achieve int+Time]



Friends

        functions---same access privileges as a member function of the class | can use class op inside[lec11]

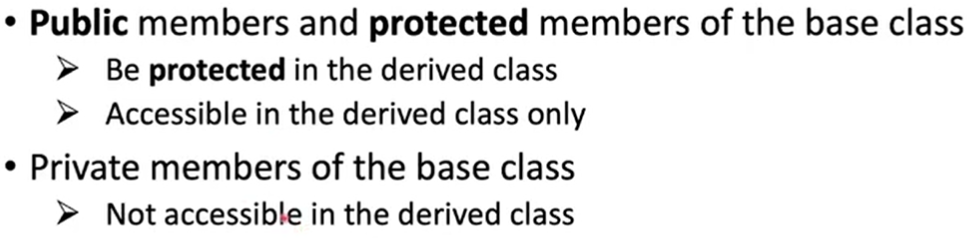
class——friend class class2::all member functions of class ClassTwo have the right to access the private and protected class members of ClassOne

        member functions

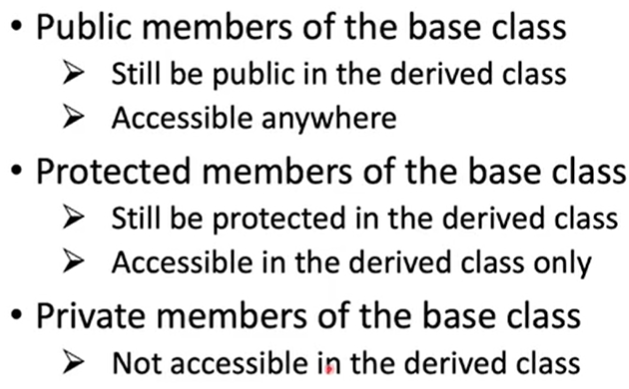
Dynamic memory&Class

Templates

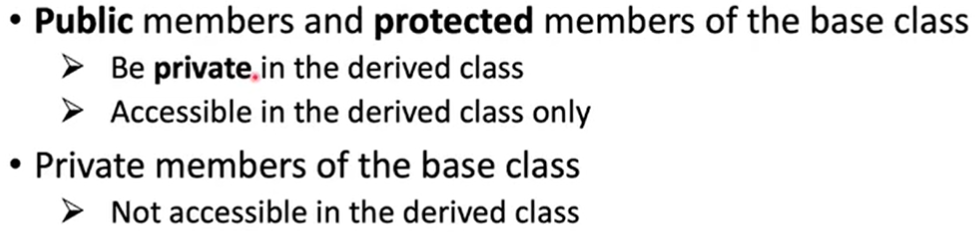
**Protected**



**Public**



**Private**

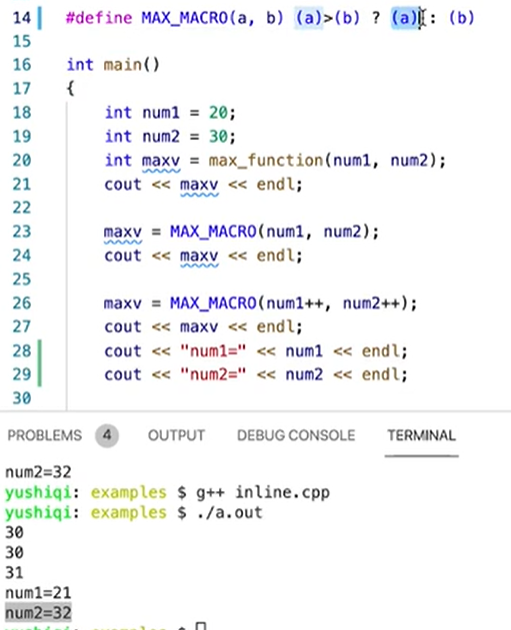


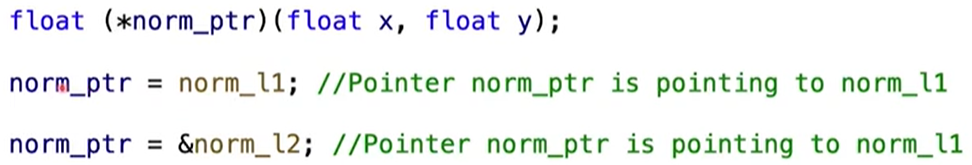
**Inheritance**

Needs virtual destructor!

**Exception**

**Function**

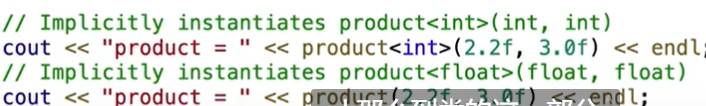
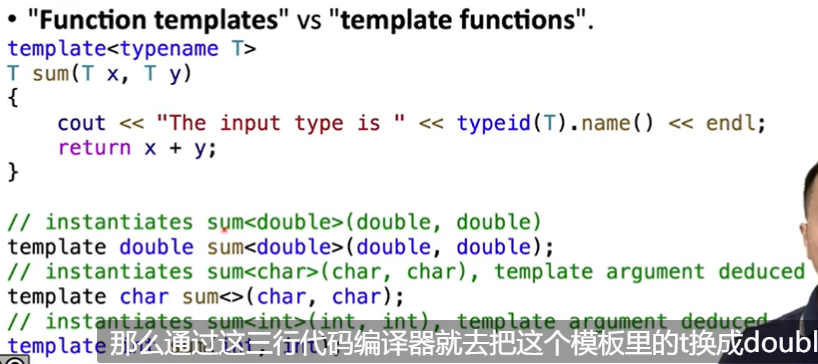
Inline/macromacro must use ()! Defined in class:auto inline

pointer[lec 6]

an augument pass to another function(e.g. qsort)

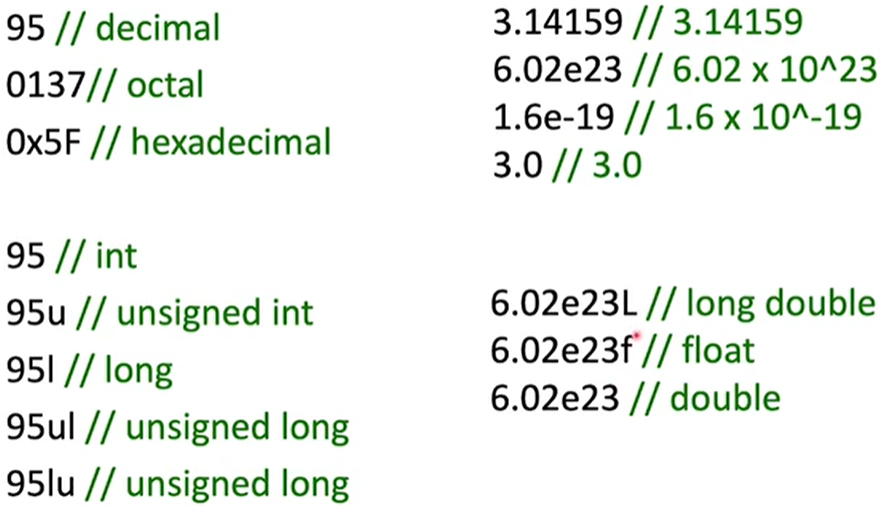
default arguments

overloading

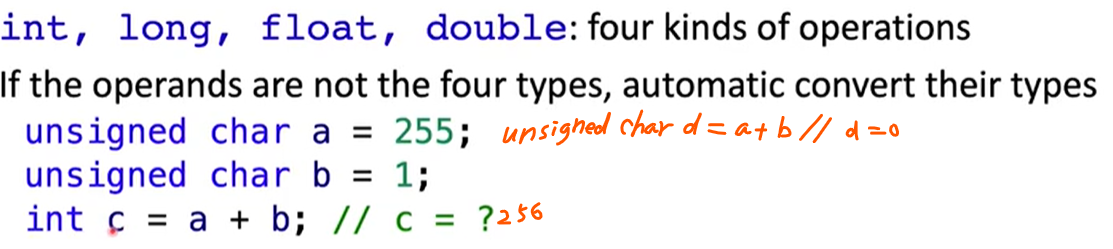
template&specialization&Instantiations[lab7] 

extern/static

**Arithmetic**



a<<b a左移b位



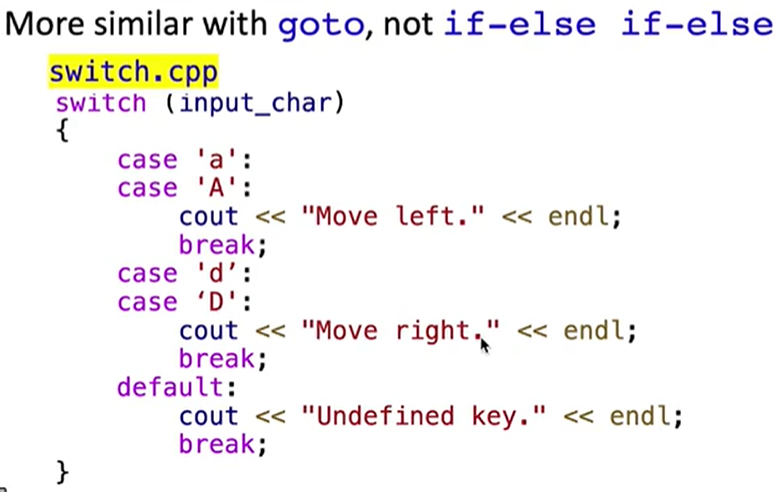
**Branch**

If else

while

Goto

Switch



**Cpp**

cin .get (remember the omitted)

   .good[lec5 p39][lab6 e3]

  [lab3]

cout .setf  .width  .fill  .precision

                 [lab2 e1]

**C**

scanf("%xx",&a)

getchar(used to omit \n, " ")  gets(&a)

printf

puts(&a) [lab6 e1]

**IO**

