

**Lecture Five** 

# **Arrays, Pointers and References**

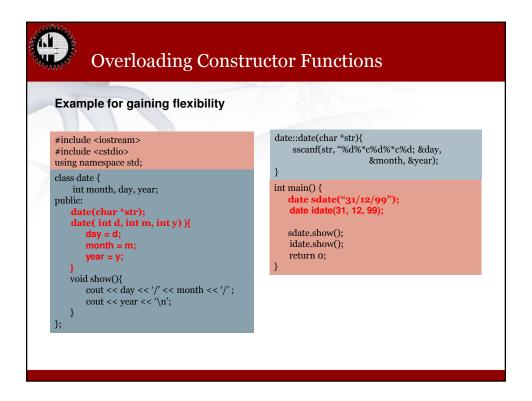
Ref: Herbert Schildt, Teach Yourself C++, Third Edn (Chapter 5)

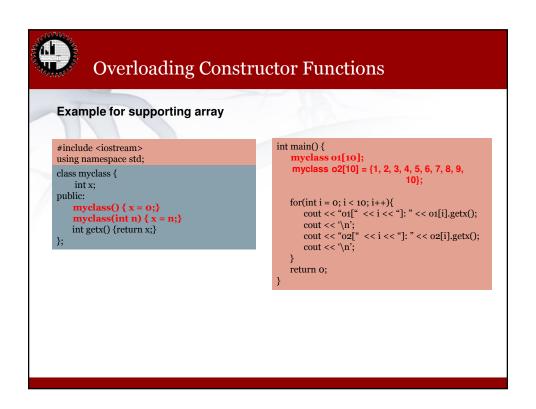
© **Dr. M. Mahfuzul Islam** Professor, Dept. of CSE, BUET



### **Overloading Constructor Functions**

- > It is common to overload a class's constructor function.
- > It is not possible to overload a destructor function.
- > Three main reasons to overload constructor function:
  - to gain flexibility,
  - to support arrays and
  - to create copy constructors.
- > If a program attempts to create an object for which no matching constructor is found, a compile-time error occurs.







## Creating and Using copy constructor

>Problems can occur when an object is passed to or returned from a function. "copy constructor" is one of the solutions.

>There are two distinct situations for assigning one object to anotherassignment and initialization. The copy constructor only applies to initializations. It does not apply to assignments.

>Common form of assignment:

classname (const classname &object){}



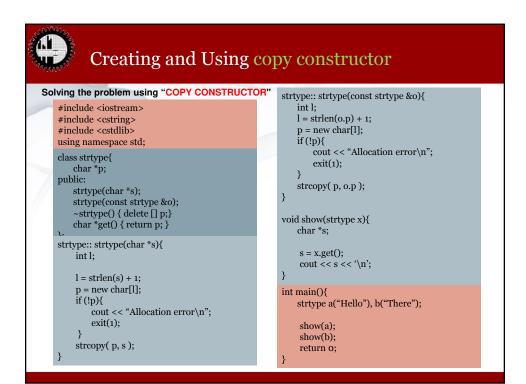
### Creating and Using copy constructor

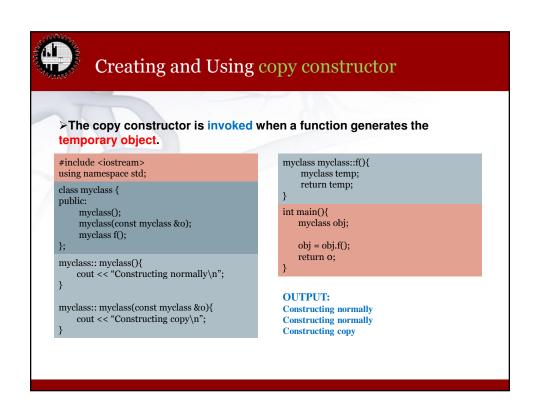
```
#include <iostream>
#include <cstring>
#include <cstdlib>
using namespace std;
class strtype{
   char *p;
public:
   strtype(char *s);
    ~strtype() { delete [] p;}
    char *get() { return p; }
strtype:: strtype(char *s){
    l = strlen(s) + 1;
     p = new char[l];
    if (!p){
        cout << "Allocation error\n";</pre>
        exit(1);
    strcopy(p, s);
```

```
void show(strtype x){
    char *s;

    s = x.get();
    cout << s << '\n';
}
int main(){
    strtype a("Hello"), b("There");
    show(a);
    show(b);
    return o;
}</pre>
```

What is the problem of the program?







### The Overload Anachronism

>When C++ was first invented, the keyword overload was used to create an overloaded function.

>Overload is obsolete now and no longer supported by modern C++ compilers.

#### The general form of overload

overload func-name;

Overloading a function called timer():

overload timer;



### **Using Default Arguments**

- >The defaults can be specified either in function prototype or in its definition if the definition precedes the function's first use.
- The defaults cannot be specified in both the prototype and the definition.
- >All default parameters must be to the right of any parameters that do not have defaults.
- > Default arguments must be constants or global variables. The cannot be local variables or other parameters.

