MA122 - Computer Programming and Applications

Indian Institute of Space Science and Technology

March 30, 2017

Lecture 25

example


```
1  #include <iostream>
2
3  using namespace std;
4
5  class Line {
6  public:
7    void setLength( double len );
8    double getLength( void );
9    Line(); // This is the constructor
10
11 private:
12   double length;
13 };
```

example


```
1
2  // Member functions definitions including constructor
3  Line::Line(void) {
4    cout << "Object is being created" << endl;
5  }
6
7  void Line::setLength( double len ) {
8    length = len;
9  }
10
11 double Line::getLength( void ) {
12   return length;
13 }
```

example


```
1  // Main function for the program
2  int main( ) {
3    Line line;
4
5    // set line length
6    line.setLength(6.0);
7    cout << "Length of line : " << line.getLength() <<
         endl;
8
9    return 0;
10 }
```

Lecture 25

example


```
1  #include <iostream>
2
3  using namespace std;
4
5  class Line {
6  public:
7    void setLength( double len );
8    double getLength( void );
9    Line();  // This is the constructor declaration
10   ~Line(); // This is the destructor: declaration
11
12 private:
13   double length;
14 };
```

example


```
1  // Member functions definitions including constructor
2  Line::Line() {
3    cout << "Object is being created" << endl;
4  }
5  Line::~Line() {
6    cout << "Object is being deleted" << endl;
7  }
8
9  void Line::setLength( double len ) {
10   length = len;
11 }
12 double Line::getLength( void ) {
13   return length;
14 }
```

example


```
1 // Main function for the program
2 int main( ) {
3   Line line;
4
5   // set line length
6   line.setLength(6.0);
7   cout << "Length of line : " << line.getLength() <<
        endl;
8   return 0;
9 }
```

example


```
1  #include<iostream>
2  using namespace std;
3  class Marks
4    {
5    public:
6      int maths;
7      int science;
8      //constructor
9      Marks() {
10       cout << "Inside Constructor"<<endl;
11       cout << "C++ Object created"<<endl;
12     }
13       //Destructor
14     ~Marks() {
15       cout << "Inside Destructor"<<endl;
16       cout << "C++ Object destructed"<<endl;
17     }
18   };
```

example


```
1  int main( )
2  {
3    {
4      Marks m1;
5    }
6
7    cout<<"Hello World !!" <<endl;
8
9
10   return 0;
11 }
```

Lecture 25

indirect membership operator


```
1  #include <iostream>
2  struct antarctica_years_end {
3      int year;
4
5  };
6  int main()
7  {
8      antarctica_years_end s01, s02, s03;
9      s01.year = 1998;
10     antarctica_years_end * pa = &s02;
11     pa->year = 1999;
12     std::cout << pa->year << std::endl;
13     antarctica_years_end trio[3]; // array of 3
           structures trio[0].year = 2003;
14     trio[0].year=s01.year;
15     std::cout << trio->year << std::endl;
16     return 0;
17 }
```

Lecture 25

example


```
1  #include <iostream>
2
3  using namespace std;
4
5  class Box {
6  public:
7      // Constructor definition
8      Box(double l = 2.0, double b = 2.0, double h =
           2.0) {
9          cout <<"Constructor called." << endl;
10         length = l;
11         breadth = b;
12         height = h;
13     }
14
15     double Volume() {
16         return length * breadth * height;
17     }
```

example


```
1
2
3       int compare(Box box) {
4           return this->Volume() > box.Volume();
5       }
6
7   // -> : indirect membership operator
8   private:
9       double length;    // Length of a box
10      double breadth;   // Breadth of a box
11      double height;    // Height of a box
12  };
```

example


```
1
2  int main(void) {
3      Box Box1(3.3, 1.2, 1.5); // Declare box1
4      Box Box2(8.5, 6.0, 2.0); // Declare box2
5
6      if(Box1.compare(Box2)) {
7          cout << "Box2 is smaller than Box1" <<endl;
8      } else {
9          cout << "Box2 is equal to or larger than Box1"
               <<endl;
10     }
11
12     return 0;
13 }
```