Below is a set of code for class Integer.

1. use pointer: change "int value;" to "int\* value;"

2. Make your class “pointer safe” by adding the following

 o Copy constructor  
 o Destructor  // Integer.h

#ifndef INTEGER\_H

#define INTEGER\_H

class Integer

{

 public:

 Integer() {value = 0;}

 Integer( int intVal );

 int getInteger() const;

 void setInteger( int newInteger );

 Integer& operator=( const Integer& rhInteger );

 private:

 int value; // This line will be removed…

 // int\* value; // and replaced with this one…

 // NOTE: You will only use ONE attribute

};

#endif

// Integer.cpp

#include “integer.h”

Integer::Integer( int intVal )

{

 value = intVal;

}

int Integer::getInteger() const

{

 return value;

}

void Integer::setInteger( int newInteger )

{

 value = newInteger;

}

Integer& Integer::operator=( const Integer& rhInt )

{

 value = rhInt.value;

 return \*this;

}

// IntegerTest.cpp

#include <iostream>

#include <cstdlib>

#include “integer.h”

using namespace std;

void displayInteger( char\* str, Integer intObj )

 {

 cout << str << “ is “ << intObj.getInteger() << endl;

 }

int main( int argc, char\* argv[] )

{

 Integer intVal1;

 Integer intVal2(10);

 displayInteger( “intVal1”, intVal1 );

 displayInteger( “intVal2”, intVal2 );

 intVal1 = intVal2;

 displayInteger( “intVal1”, intVal1 );

 return EXIT\_SUCCESS;

}