## Step-by-step walkthrough for example on page 53 of the lecture notes: C++ Class

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
  y.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
  x.print();
  x.add1(y).add1(y).print();
  x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
  p->print();
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
  z.print();
  z.add3(y).add3(y).print();
  z.print();
  return 0;
```

```
class Complex /* File: complex.h */
 private:
    float real; float imag;
 public:
   Complex(float r, float i) { real = r; imag = i; }
   void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this;
    Complex& add3(const Complex& x)
    // Return by reference
      real += x.real; imag += x.imag;
      return (*this);
};
```

```
class Complex /* File: complex.h */
                                                       this
                                           3
                                                  4
 private:
    float real; float imag;
 public:
    Complex(float r, float i) { real = r; imag = i; }
   void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this:
    Complex& add3(const Complex& x)
    // Return by reference
      real += x.real; imag += x.imag;
      return (*this):
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
  v.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
  x.print();
  x.add1(y).add1(y).print();
  x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
  p->print();
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
  z.print();
  z.add3(y).add3(y).print();
  z.print();
  return 0;
```

```
class Complex /* File: complex.h */
                                                       this
 private:
    float real; float imag;
 public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this;
                                                               Output
                                                               (3,4)
    Complex& add3(const Complex& x)
    // Return by reference
      real += x.real; imag += x.imag;
      return (*this);
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
  y.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
  x.print();
  x.add1(y).add1(y).print();
  x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
  p->print();
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                                  Output
  z.print();
                                                                  (3,4)
  z.add3(y).add3(y).print();
                                                                  Return by value
  z.print();
  return 0;
}
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
 y.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
 x.print();
 x.add1(y).add1(y).print();
 x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
  p->print();
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                                  Output
  z.print();
                                                                 (3,4)
  z.add3(y).add3(y).print();
                                                                 Return by value
  z.print();
  return 0;
```

```
class Complex /* File: complex.h */
                                                  2
 private:
   float real; float imag;
  public:
   Complex(float r, float i) { real = r; imag = i; }
   void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
   Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
                                                                imag
                                                                      2
      return (*this);
   Complex* add2(const Complex& x)
                                                       this
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this;
                                                               Output
                                                               (3,4)
   Complex& add3(const Complex& x)
                                                               Return by value
    // Return by reference
      real += x.real; imag += x.imag;
      return (*this);
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
 v.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
  x.print();
 x.add1(y).add1(y).print();
 x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
  p->print();
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                                 Output
 z.print();
                                                                 (3,4)
  z.add3(y).add3(y).print();
                                                                 Return by value
 z.print();
 return 0;
```

```
class Complex /* File: complex.h */
  private:
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
                                                                      2
                                                                imag
      return (*this);
    Complex* add2(const Complex& x)
                                                       this
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this:
                                                               Output
                                                               (3,4)
    Complex& add3(const Complex& x)
                                                               Return by value
    // Return by reference
                                                               (1,2)
      real += x.real; imag += x.imag;
      return (*this):
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
 y.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
 x.print();
  x.add1(y).add1(y).print();
 x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
  p->print();
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                                  Output
  z.print();
                                                                 (3,4)
  z.add3(y).add3(y).print();
                                                                 Return by value
  z.print();
                                                                 (1,2)
  return 0;
```

```
Nickname at add1
class Complex /* File: complex.h */
  private:
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
                                                                imag
                                                                       2
      return (*this);
                                                     this
    Complex* add2(const Complex& x)
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this:
                                                                Output
                                                                (3,4)
    Complex& add3(const Complex& x)
                                                                Return by value
    // Return by reference
                                                                (1,2)
      real += x.real; imag += x.imag;
      return (*this);
};
                                                              Nickname at add1
class Complex /* File: complex.h */
  private:
    float real; float imag;
```

```
Nickname at add1
class Complex /* File: complex.h */
 private:
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
                                                                imag
      return (*this);
                                                     this
    Complex* add2(const Complex& x)
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this:
                                                                Output
                                                                (3,4)
    Complex& add3(const Complex& x)
                                                                Return by value
    // Return by reference
                                                                (1,2)
      real += x.real; imag += x.imag;
      return (*this);
};
```

```
public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
                                                                imag
                                                                       6
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this:
                                                                Output
                                                                (3,4)
    Complex& add3(const Complex& x)
                                                                Return by value
    // Return by reference
                                                                (1,2)
      real += x.real; imag += x.imag;
      return (*this):
                                           temp
                         Generated by return (*this)
};
```

```
#include <iostream> /* File: complex-test.cpp */
   using namespace std;
   #include "complex.h"
   int main()
     Complex y(3, 4);
     y.print();
     cout << endl << "Return by value" << endl;</pre>
     Complex x(1, 2);
     x.print();
temp x.add1(y).print();
     x.print();
     cout << endl << "Return its pointer by value" << endl;</pre>
     Complex* p = x.add2(y);
     p->print();
     cout << endl << "Return by reference" << endl;</pre>
     Complex z(1, 2);
                                                                    Output
     z.print();
                                                                    (3,4)
     z.add3(y).add3(y).print();
                                                                    Return by value
     z.print():
                                                                    (1,2)
     return 0;
                                               temp
```

```
Nickname at add1
class Complex /* File: complex.h */
  private:
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
                                                                imag
                                                                       6
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this:
                                                                Output
                                                                (3,4)
    Complex& add3(const Complex& x)
                                                                Return by value
    // Return by reference
                                                                (1,2)
      real += x.real; imag += x.imag;
      return (*this);
                                           temp
                                     this
};
                                                              Nickname at add1
class Complex /* File: complex.h */
  private:
```

```
Nickname at add1
class Complex /* File: complex.h */
 private:
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
                                                                imag
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this:
                                                                Output
                                                               (3,4)
    Complex& add3(const Complex& x)
                                                                Return by value
    // Return by reference
                                                               (1,2)
      real += x.real; imag += x.imag;
      return (*this);
                                           temp
                                                         10
                                                  imag
                                     this
};
```

```
float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
                                                                       6
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this:
                                                                Output
                        Generated by return (*this)
                                                                (3,4)
                                          temp2
    Complex& add3(const Complex& x)
                                                         10
                                                   imag
                                                                Return by value
    // Return by reference
                                                                (1,2)
      real += x.real; imag += x.imag;
      return (*this):
                                           temp
                                                         10
                                                   imag
                                     this
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
 y.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
 x.print(); temp2
- x.add1(y).add1(y).print();
  x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
  p->print();
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                                 Output
  z.print();
                                                                 (3,4)
  z.add3(y).add3(y).print();
                                                                 Return by value
  z.print():
                                                                 (1,2)
  return 0;
                                           temp2
                                                    imag
```

```
Nickname at add1
class Complex /* File: complex.h */
  private:
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
                                                                imag
                                                                       6
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this;
                                                                Output
                                                                (3,4)
    Complex& add3(const Complex& x)
                                                                Return by value
    // Return by reference
                                                                (1,2)
                                            this
                                                                (7,10)
      real += x.real; imag += x.imag;
                                                   real
      return (*this);
                                          temp2
                                                         10
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
 v.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
 x.print();
 x.add1(y).add1(y).print();
 x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
  p->print();
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                                 Output
 z.print();
                                                                 (3,4)
  z.add3(y).add3(y).print();
                                                                 Return by value
 z.print();
                                                                 (1,2)
 return 0;
                                                                 (7,10)
```

```
class Complex /* File: complex.h */
  private:
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
      real += x.real; imag += x.imag;
                                                                      6
                                                                imag
      return (*this);
    Complex* add2(const Complex& x)
                                                       this
    // Return by value using pointer
      real += x.real; imag += x.imag;
      return this:
                                                               Output
                                                               (3,4)
    Complex& add3(const Complex& x)
                                                               Return by value
    // Return by reference
                                                               (1,2)
                                                               (7,10)
                                                               (4,6)
      real += x.real; imag += x.imag;
      return (*this):
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
 y.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
 x.print();
 x.add1(y).add1(y).print();
 x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
                                                        Output
  p->print();
                                                        (3,4)
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                        Return by value
                                                        (1, 2)
  z.print();
                                                         (7,10)
  z.add3(y).add3(y).print();
                                                        (4,6)
  z.print();
                                                        Return its pointer by value
  return 0;
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
  v.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
  x.print();
  x.add1(y).add1(y).print();
  x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
                                                        Output
  p->print();
                                                        (3,4)
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                        Return by value
  z.print();
                                                        (1,2)
                                                        (7,10)
  z.add3(y).add3(y).print();
                                                        (4,6)
  z.print();
                                                        Return its pointer by value
  return 0:
```

```
Nickname at add1
class Complex /* File: complex.h */
 private:
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                            Χ
      real += x.real; imag += x.imag;
                                                                 imag
      return (*this);
                                                      this
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1,2)
                                                       (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
      real += x.real; imag += x.imag;
      return (*this);
};
```

```
Nickname at add1
class Complex /* File: complex.h */
  private:
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                            Х
      real += x.real; imag += x.imag;
                                                                 imag 10
      return (*this);
                                                      this
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1, 2)
                                                       (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
      real += x.real; imag += x.imag;
      return (*this):
};
```

```
Nickname at add1
class Complex /* File: complex.h */
  private:
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                                  real
                                                            Х
      real += x.real; imag += x.imag;
                                                                 imag 10
      return (*this);
                                                      this
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1, 2)
                                                       (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
      real += x.real; imag += x.imag;
      return (*this):
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
  v.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
  x.print();
                                                                  imag 10
  x.add1(y).add1(y).print();
  x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
                                                        Output
  p->print();
                                                        (3,4)
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                        Return by value
  z.print();
                                                        (1,2)
                                                        (7,10)
  z.add3(y).add3(y).print();
                                                        (4,6)
  z.print();
                                                        Return its pointer by value
  return 0:
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
 v.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
 x.print();
                                                                  imag 10
 x.add1(y).add1(y).print();
 x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
                                                        Output
  p->print();
                                                        (3,4)
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                        Return by value
 z.print();
                                                        (1,2)
                                                        (7,10)
  z.add3(y).add3(y).print();
                                                        (4,6)
 z.print();
                                                        Return its pointer by value
 return 0:
```

```
class Complex /* File: complex.h */
  private:
                                                                 imag
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                            Х
      real += x.real; imag += x.imag;
                                                                       10
                                                                 imag
      return (*this);
                                                   this
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1, 2)
                                                       (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
                                                       (7,10)
      real += x.real; imag += x.imag;
      return (*this):
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
  Complex y(3, 4);
 y.print();
  cout << endl << "Return by value" << endl;</pre>
                                                                   real
  Complex x(1, 2);
                                                              Х
 x.print();
                                                                          10
                                                                   imag
 x.add1(y).add1(y).print();
 x.print():
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
                                                         Output
  p->print();
                                                         (3,4)
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2);
                                                         Return by value
                                                         (1, 2)
  z.print();
                                                         (7,10)
  z.add3(y).add3(y).print();
                                                         (4,6)
  z.print();
                                                         Return its pointer by value
  return 0;
                                                        (7,10)
                                                         Return by reference
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
                                                                   imag
int main()
                                                real
                                           Z
  Complex y(3, 4);
  v.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
  x.print();
                                                                   imag 10
  x.add1(y).add1(y).print();
  x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
                                                         Output
  p->print();
                                                         (3,4)
  cout << endl << "Return by reference" << endl:</pre>
  Complex z(1, 2);
                                                         Return by value
  z.print();
                                                         (1,2)
                                                         (7,10)
  z.add3(y).add3(y).print();
                                                         (4,6)
  z.print();
                                                         Return its pointer by value
  return 0:
                                                         (7,10)
                                                         Return by reference
```

```
class Complex /* File: complex.h */
                                      this
                                                     1
                                               real
                                          Z
                                                            V
  private:
                                                      2
    float real; float imag;
                                     2
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                            Χ
      real += x.real; imag += x.imag;
                                                                 imag 10
      return (*this):
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1,2)
                                                       (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
                                                       (7,10)
      real += x.real; imag += x.imag;
                                                       Return by reference
      return (*this);
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
                                                real
                                           Z
                                                       2
  Complex y(3, 4);
  y.print();
  cout << endl << "Return by value" << endl;</pre>
                                                                          7
  Complex x(1, 2);
  x.print();
                                                                         10
                                                                   imag
  x.add1(y).add1(y).print();
  x.print():
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
                                                         Output
  p->print():
                                                        (3,4)
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2):
                                                        Return by value
                                                        (1, 2)
  z.print();
                                                         (7,10)
  z.add3(y).add3(y).print();
                                                        (4,6)
  z.print();
                                                        Return its pointer by value
  return 0;
                                                        (7,10)
                                                        Return by reference
```

```
class Complex /* File: complex.h */
                                       this
                                          Z
                                                             У
  private:
                                                      2
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                                  real
                                                             Х
      real += x.real; imag += x.imag;
                                                                        10
                                                                 imag
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1, 2)
                                                        (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
                                                       (7,10)
      real += x.real; imag += x.imag;
                                                       Return by reference
      return (*this):
                                                       (1, 2)
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
                                                                   imag
                                                       1
int main()
                                                real
                                           Z
                                                       2
  Complex y(3, 4);
  v.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
  x.print();
                                                                   imag 10
  x.add1(y).add1(y).print();
  x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
                                                        Output
  p->print();
                                                        (3,4)
  cout << endl << "Return by reference" << endl:</pre>
  Complex z(1, 2);
                                                        Return by value
  z.print();
                                                        (1,2)
                                                        (7,10)
  z.add3(y).add3(y).print();
                                                        (4,6)
  z.print();
                                                        Return its pointer by value
  return 0:
                                                        (7,10)
                                                        Return by reference
```

```
class Complex /* File: complex.h */
                                          Z
 private:
                                                     2
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                            Χ
      real += x.real; imag += x.imag;
                                                                 imag 10
      return (*this):
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1,2)
                                                       (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
                                                       (7,10)
      real += x.real: imag += x.imag:
                                                       Return by reference
      return (*this);
                                                       (1,2)
};
```

```
class Complex /* File: complex.h */
                                               real
                                          Z
  private:
                                                      6
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                            Х
      real += x.real; imag += x.imag;
                                                                 imag 10
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1, 2)
                                                       (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
                                                       (7,10)
      real += x.real; imag += x.imag;
                                                       Return by reference
      return (*this):
                                                       (1, 2)
};
```

```
Generated by return (*this)
                                                 temp3
class Complex /* File: complex.h */
                                               real
                                          Z
  private:
                                                      6
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                                  real
                                                             Х
      real += x.real; imag += x.imag;
                                                                 imag 10
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                        Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                        (1, 2)
                                                        (7,10)
    Complex& add3(const Complex& x)
                                                        (4,6)
    // Return by reference
                                                        Return its pointer by value
                                                       (7,10)
      real += x.real; imag += x.imag;
                                                       Return by reference
      return (*this):
                                                       (1, 2)
};
```

```
#include <iostream> /* File: complex-test.cpp */
   using namespace std;
    #include "complex.h"
                                                      temp3
                                                                      imag
    int main()
                                                    real
                                               Z
      Complex y(3, 4);
      v.print();
      cout << endl << "Return by value" << endl;</pre>
      Complex x(1, 2);
      x.print();
                                                                      imag 10
      x.add1(y).add1(y).print();
      x.print();
      cout << endl << "Return its pointer by value" << endl;</pre>
      Complex* p = x.add2(y);
                                                            Output
      p->print();
                                                            (3,4)
      cout << endl << "Return by reference" << endl:</pre>
      Complex z(1, 2);
                                                            Return by value
      z.print();
                                                            (1, 2)
                                                            (7,10)
temp3 z.add3(y).print();
                                                            (4,6)
      z.print();
                                                            Return its pointer by value
      return 0:
                                                            (7,10)
                                                            Return by reference
```

```
class Complex /* File: complex.h */
                                          Z
 private:
                                                     6
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                            Χ
      real += x.real; imag += x.imag;
                                                                 imag 10
      return (*this):
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1,2)
                                                       (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
                                                       (7,10)
      real += x.real: imag += x.imag:
                                                       Return by reference
      return (*this);
                                                       (1,2)
};
```

```
class Complex /* File: complex.h */
                                          Z
  private:
                                                     10
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                            Х
      real += x.real; imag += x.imag;
                                                                 imag 10
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1, 2)
                                                       (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
                                                       (7,10)
      real += x.real; imag += x.imag;
                                                       Return by reference
      return (*this):
                                                       (1, 2)
};
```

```
Generated by return (*this)
                                                 temp4
class Complex /* File: complex.h */
                                               real
                                          Z
  private:
                                                     10
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                                  real
                                                             Х
      real += x.real; imag += x.imag;
                                                                 imag 10
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                        Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                        (1, 2)
                                                        (7,10)
    Complex& add3(const Complex& x)
                                                        (4,6)
    // Return by reference
                                                        Return its pointer by value
                                                       (7,10)
      real += x.real; imag += x.imag;
                                                        Return by reference
      return (*this):
                                                       (1, 2)
};
```

```
#include <iostream> /* File: complex-test.cpp */
   using namespace std;
                                                                              3
    #include "complex.h"
                                                      temp4
                                                                       imag
    int main()
                                                    real
                                               Z
                                                          10
      Complex y(3, 4);
      v.print();
      cout << endl << "Return by value" << endl;</pre>
      Complex x(1, 2);
      x.print();
                                                                       imag 10
      x.add1(y).add1(y).print();
      x.print();
      cout << endl << "Return its pointer by value" << endl;</pre>
      Complex* p = x.add2(y);
                                                            Output
      p->print();
                                                            (3,4)
      cout << endl << "Return by reference" << endl:</pre>
      Complex z(1, 2);
                                                            Return by value
      z.print();
                                                            (1, 2)
                                                            (7,10)
temp4 z.add3(y).add3(y).print();
                                                            (4,6)
      z.print();
                                                            Return its pointer by value
      return 0:
                                                            (7,10)
                                                            Return by reference
                                                            (1,2)
```

```
class Complex /* File: complex.h */
                                               real
                                          Z
 private:
                                                     10
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                             Χ
      real += x.real; imag += x.imag;
                                                                 imag 10
      return (*this):
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                       Output
                                                       (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                       Return by value
                                                       (1,2)
                                                       (7,10)
    Complex& add3(const Complex& x)
                                                       (4,6)
    // Return by reference
                                                       Return its pointer by value
                                                       (7,10)
      real += x.real: imag += x.imag:
                                                       Return by reference
      return (*this);
                                                       (1, 2)
                                                       (7, 10)
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
int main()
                                                real
                                           Z
                                                       10
  Complex y(3, 4);
  y.print();
  cout << endl << "Return by value" << endl;</pre>
                                                                          7
  Complex x(1, 2);
  x.print();
                                                                         10
                                                                   imag
  x.add1(y).add1(y).print();
  x.print():
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
                                                         Output
  p->print():
                                                        (3,4)
  cout << endl << "Return by reference" << endl;</pre>
  Complex z(1, 2):
                                                        Return by value
                                                        (1, 2)
  z.print();
                                                         (7,10)
  z.add3(y).add3(y).print();
                                                        (4,6)
  z.print();
                                                        Return its pointer by value
  return 0;
                                                        (7,10)
                                                        Return by reference
                                                        (1, 2)
                                                        (7, 10)
```

```
class Complex /* File: complex.h */
                                       this
                                                                  real
                                           Z
  private:
                                                      10
    float real; float imag;
  public:
    Complex(float r, float i) { real = r; imag = i; }
    void print() { cout << "( " << real << " , " << imag << " )" << endl; }</pre>
    Complex add1(const Complex& x) // Return by value
                                                                  real
                                                             Х
      real += x.real; imag += x.imag;
                                                                        10
                                                                  imag
      return (*this);
    Complex* add2(const Complex& x)
    // Return by value using pointer
                                                        Output
                                                        (3,4)
      real += x.real; imag += x.imag;
      return this:
                                                        Return by value
                                                        (1, 2)
                                                        (7,10)
    Complex& add3(const Complex& x)
                                                        (4,6)
    // Return by reference
                                                        Return its pointer by value
                                                        (7,10)
      real += x.real; imag += x.imag;
                                                        Return by reference
      return (*this):
                                                        (1, 2)
                                                        (7, 10 )
                                                        (7, 10)
};
```

```
#include <iostream> /* File: complex-test.cpp */
using namespace std;
#include "complex.h"
                                                                  imag
int main()
                                                      7
                                           Z
                                               imag
                                                      10
  Complex y(3, 4);
  y.print();
  cout << endl << "Return by value" << endl;</pre>
  Complex x(1, 2);
  x.print();
                                                                  imag 10
  x.add1(y).add1(y).print();
  x.print();
  cout << endl << "Return its pointer by value" << endl;</pre>
  Complex* p = x.add2(y);
                                                        Output
  p->print();
  cout << endl << "Return by reference" << endl;</pre>
                                                        (3,4)
  Complex z(1, 2);
                                                        Return by value
                                                       (1,2)
(7,10)
(4,6)
  z.print();
  z.add3(y).add3(y).print();
  z.print();
                                                        Return its pointer by value
  return 0;
                                                        (7,10)
                                                        Return by reference
                                                        (1,2)
                                                        (7, 10)
                                                        (7, 10)
```