ECE 30862 Fall 2017, Second Exam

DO NOT START WORKING ON THIS UNTIL TOLD TO DO SO. LEAVE IT ON THE DESK.

THE LAST PAGE IS THE ANSWER SHEET. TEAR IT OFF AND PUT ALL ANSWERS THERE. TURN IN BOTH PARTS OF THE TEST WHEN FINISHED.

You have until 9:00PM to take this exam. There are 50 questions and each is worth two points. After taking the test, turn in both the test and the answer sheet.

Your exam should have this sheet, 7 pages with 50 questions, and the answer sheet. As soon as the test begins, check that your exam is complete and *let Prof. Midkiff know immediately if it does not.*

This exam is open book, open notes, but absolutely no electronics. If you have a question, please ask for clarification. If the question is not resolved, state on the test whatever assumptions you need to make to answer the question, and answer it under those assumptions.

If a statement is illegal, assume it is not executed when answering other questions in the test.

Check the front board occasionally for corrections.

I have neither given nor received help during this exam from any other person or electronic source, and I understand that if I have I will be guilty of cheating and will fail the exam and perhaps the course, at the instructor's discretion.

Name (must be signed to be graded):

Name:

Last four digits of your ID:

C++Questions. For each question **Q1 - Q9** below answer what is printed by the commented line on your answer sheet. If a runtime or compile time error, answer "Err". If the statement is legal and nothing is printed, answer "Ok". If the statement is illegal, execute the remainder of the program as if the illegal statement did not exist.

A.h

```
class A {
class A {
public:
   static int count; // Q1 Assume the line
                     // at Q2 does not exist.
   static int count = 0; // Q2 Assume
                     // the line at Q1 does not
                     // exist:

   int counter;
   A( );
   static void incr( );
   virtual void print( );
};
```

A.cpp

```
int A::count = 0; // Q3
int A::counter = 0; // Q4

void A::incr( ) {
   count++; // Q5
   counter++; // Q6
}

void A::print( ) {
   std::cout << count << std::endl; // Q7
   std::cout << counter << std::endl; // Q8
}
```

E.h

```
class E {
public:
   std::string msg;
   E(std::string);
   virtual void print( );
};
```

E.cpp

```
E::E(std::string m) {
   A::count++;
   msg = m;
}

void E::print( ) {
   std::cout << "E: " << msg << " ";
   std::cout << A::count << std::endl;
}
```

main.cpp

```
void foo(int j) {
   if (j < 0) throw E("Err" );
   if (j == 0) throw 1;
}

int main( ) {
   for (int i = -1; i < 1; i++) { // A
      try {
         foo(i);
      }
      catch (E e) {e.print( );}
      catch (int i) {std::cout << i << std::endl;}
   }

   std::cout << A::count << std::endl; // Q9
}
```

Q10. What is printed during the entire execution of the loop at the statement marked with **A**?

C++ questions. For each question **Q11 - Q15** below answer what is printed by the commented line on your answer sheet. If a runtime or compile time error, answer "Err". If the statement is legal and nothing is printed, answer "Ok". If the statement is illegal, execute the remainder of the program as if the illegal statement did not exist.

Base.h

```
class Base {
public:
  int* x;
  Base( );
  virtual ~Base( );
};
```

Base.cpp

```
Base::Base( ) {
  std::cout << "Base" << std::endl;
  x = new int[3];
  x[0]= 0; x[1] = 1; x[2] = 2;
}

Base::~Base( ) {
  std::cout << "~Base" << std::endl;
  delete x;
}
```

Q15. When exiting the main routine, how many times is the array freed that is allocated when the object held in the variable "a" is constructed?

A.h

```
class A : public Base {
public:
  A( );
  virtual ~A( );
};
```

A.cpp

```
A::A( ) {
  std::cout << "A" << endl;
}

A::~A( ) {
  cout << "~A" << std::endl;
};
```

main.cpp

```
int main( ) {
  A a; // Q11
  Base b; // Q12

  b = a;
  b.x[1] = -1;

  std::cout << a.x[1] << std::endl; // Q13
  std::cout << b.x[1] << std::endl; // Q14

}
```

C++ questions. For each question **Q16 - Q22** below answer what is printed by the commented line on your answer sheet. If a runtime or compile time error, answer "Err". If the statement is legal and nothing is printed, answer "Ok". If the statement is illegal, execute the remainder of the program as if the illegal statement did not exist.

Int.h

```
class Int {
public:
  Int( );
  Int(int);
  Int(const Int&);
  virtual ~Int( );
  Int operator=(const Int&);
  Int operator+(const Int);
  Int operator-(const Int&);
  Int operator-( ); // A
  std::ostream& operator>>(std::ostream&);
  friend Int operator-(const Int&); // B
  friend std::ostream& operator<<(
      std::ostream&, const Int&);  // C
private:
  int val;
};

Int operator-(const Int&); // D

std::ostream& operator<<(std::ostream&, const
Int&);
```

Int.cpp

```
Int::Int( ) {
  val = 0;
}

Int::Int(int i) {
  val = i;
}

Int::Int(const Int& src) {
  val = src.val*src.val;
}


Int::~Int( ) { }

Int Int::operator=(const Int& i) {
  Int n = i;
  return n;
}

Int Int::operator+(const Int i) {
  Int n;
  n.val = this->val + i.val;
  return n;
}
```

Int.cpp continued

```
Int Int::operator-(const Int& i) {
  Int n;
  n.val = i.val - this->val;
  return n;
}

Int Int::operator-( ) {  // E
  Int n;
  n.val = -this->val;
  return n;
}

Int operator-(const Int& i) {  // F
  Int n;
  n.val = -i.val;
  return n;
}

std::ostream& Int::operator>>(std::ostream& os) {
  os << val;
  return os;
}

std::ostream& operator<<(std::ostream& os, const Int& i) {
  os << i.val;
  return os;
}

int main( ) {
  Int i1(1);
  Int i2(2);
  Int i3;
  std::cout << i1 << " " << i2 << " " << i3 << std::endl; // Q16

  std::cout << i1 << " " << i2 << " " << i3 << std::endl; // Q17
  std::cout << "i1: ";
  i1 >> std::cout; // Q18
  std::cout << std::endl;

  i3 = i1 + i2; // G
  std::cout << i1 << " " << i2 << " " << i3 << std::endl; // Q19

  Int i4 = i3 = i1 - i2;
  std::cout << i1 << " " << i2 << " "; // Q20
  std::cout  << i3 << " " << i4 << std::endl; // Q21

  i4 = -i3;
  std::cout << i3 << std::endl; // Q22
}
```

C++ question. The questions below refer to the program on the previous page.

Q23. Pick all that are true. For the two functions declared at A, B, C and D, and defined at E and F (answer all that are true)
(a) They both do the same thing and only one can legally be in the program.
(b) They both do the same thing and both can legally be in the program at the same time.
(d) They do different things and both can legally be in the program.
(e) They do different things and only one can legally be in the program.
(f) The function declared at A is legal but the one declared at D is not.
(g) The function declared at D is legal but the one declared at A is not.
(h) None of the above.

Q24. Could the overloaded "<<" be a member function? Answer T or F.

Q25. what does the *this* pointer point to when executing "std::cout << i1" in the line of Question Q21. Give the name of the variable pointed to.

The following two questions have nothing to do with the program on the previous page.

Q26. You need to keep records of all homework done. The last homework done should be the first visited when accessing the container. Accesses will be done linearly. Is a List or Vector preferred?

Q27. You have 1000 customers, with customer numbers from 0 to 999. Customers will be added to the end of the container. You need to access their records in constant

Java question. For each question **Q28 - Q32** below answer what is printed by the commented line on your answer sheet. If a runtime or compile time error, answer "Err". If the statement is legal and nothing is printed, answer "Ok". If the statement is illegal, execute the remainder of the program as if the illegal statement did not exist.

```
class B { }

class D1 extends B { }

class D2 extends D1 { }

class Main {

  void foo(int i, long l, double d) {
    System.out.println("ild");
  }

  void foo(int i, int i2, double d) {
    System.out.println("isd");
  }

  void foo(short s, int i, double d) {
    System.out.println("sid");
  }

  void foo(short s, int i) {
    System.out.println("sid");
  }

  void foo(B b, D1 d) {
    System.out.println("bd1");
  }

  void foo(D1 d1, D2 d2) {
    System.out.println("bd");
  }

  public static void main(String args[]) {

    B b = new B( );
    D1 d1 = new D1( );
    D2 d2 = new D2( );
    Main m = new Main( );
    int i = 0;
    long l = 0;
    short s  = 0;
    double d  = 0.0;
    float f  = 0.0f;
    char c = '0';

    m.foo(d1, d1); // Q28
    m.foo(b, d2); // Q29
    m.foo(c, i); // Q30
    m.foo(i, s, f); // Q31
    m.foo(s, s, f); // Q32
  }
}
```

Java question. For each question **Q33 - Q45** below answer what is printed by the commented line on your answer sheet. If a runtime or compile time error, answer "Err". If the statement is legal and nothing is printed, answer "Ok". If the statement is illegal, execute the remainder of the program as if the illegal statement did not exist.

```
class B {

  public void f1( ) {
    System.out.println("B::f1");
    f4( );
  }

  public void f2(int i2) {System.out.println("B::f2");}

  public void f3(short i3) {System.out.println("B::f3");}

  public void callf4( ) {f4( );}

  private void f4( ) {System.out.println("B::f4");}
}

class D1 extends B {

  public void f2( ) {
    System.out.println("D1::f2");
  }

  public void f3(int f3) {
    System.out.println("D1::f3");
  }

  public void f4( ) {
    System.out.println("D1::f4");
  }
}

class D2 extends D1 {

  public void f3( ) {
    System.out.println("D2::f3");
  }

  public void f4( ) {
    System.out.println("D2::f4");
  }

  public void f5( ) {
    System.out.println("D2::f5");
  }
}
```

```
class Main {

  public static void main(String args[]) {
    B b = new D1( );
    D1 d1 = new D1( );
    D1 d1_2 = new D2( );
    D2 d2 = new D2( );
    short s = 0;
    int i = 0;

    b.f3(i); // Q33
    b.f4( ); // Q34

    d1.f1( ); // Q35
    d1.f2(1); // Q36
    d1.f3(s); // Q37
    d1.f3(i); // Q38
    d1.f4( ); // Q39

    d1_2.f3(i); // Q40
    d1_2.f3( ); // Q41
    d1_2.f4( ); // Q42
    d1_2.f5( ); // Q43

    b = d2;
    d1 = d2;
    b.callf4( ); // Q44
    d1.callf4( ); // Q45


  }
}
```

Java question. For each question **Q46 - Q50** below answer what is printed by the commented line on your answer sheet. If a runtime or compile time error, answer "Err". If the statement is legal and nothing is printed, answer "Ok". If the statement is illegal, execute the remainder of the program as if the illegal statement did not exist.

```
class B {

  public void f1(B b, D d) {
    b.f2( );
    d.f2( );
  }

  private void f2( ) {System.out.println("B::f2");}
}

class D extends B {

  public D(int i) {
    val = i;
  }

  public void swap(D d1, D d2) {
    D tmp = d1;
    d1 = d2;
    d2 = tmp;
    System.out.println("d1: "+d1.val+" "+d2.val);
  }

  public void swap(R r1, R r2) {
    D tmp = r1.r;
    r1.r = r2.r;
    r2.r = tmp;
    System.out.println("r1: "+r1.r.val+", r2: "+r2.r.val);
  }

  public void f2( ) {System.out.println("D::f2");}

  public int val;
}

class R {

  public R(D ref) {
    r = ref;
  }

  public D r;
}
```

```
class Main {

  public static void main(String args[]) {
    B b = new B( );
    D d1 = new D(1);
    D d2 = new D(2);

    b.f1(d1, d1);

    d1.swap(d1, d2); // Q46
    System.out.println(d1.val+" "+d2.val); // Q47

    R r1 = new R(d1);
    R r2 = new R(d2);
    d1.swap(r1, r2); // Q48
    System.out.println(d1.val+" "+d2.val); // Q49
    d1 = r1.r;
    d2 = r2.r;
    System.out.println(r1.r.val+" "+r2.r.val); // Q50
  }
}
```

ECE 30862 Fall 2017 First Exam Answer Sheet

Name (Printed):

Name (Signed):

1.

2.

3.

4.

5.

6.

7.

8.

9.

10.

11.

12.

13.

14.

15.

16.

17.

18.

19.

20.

21.

22.

23.

24.

25.

26.

27.

28.

29.

30.

31.

32.

33.

34.

35.

36.

37.

38.

39.

40.

41.

42.

43.

44.

45.

46.

47.

48.

49.

50.