ECE 30862 Fall 2019 Third Exam Answer Sheet

Put your name above – it's worth 2.5 points!

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.

This page intentionally left almost blank

ECE 30862 Fall 2019, Test 3

DO NOT START WORKING ON THIS UNTIL TOLD TO DO SO.

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

You have until 9:00 to take this exam. Each of the 39 questions is worth 2.5 points. There are 2.5 more for putting your name on your paper. After taking the test turn in both the test and the answer sheet.

Your exam should have 11 (eleven) pages total (including this cover page, the answer sheet and one almost entire blank page). 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 answer sheet whatever assumptions you made to answer the question, and answer it under those assumptions.

Check the front board occasionally for corrections.

Programs may be given without "#include" statements for brevity. Assume all needed includes are present. "std::endl" has been left off for brevity for some statements. You may use newlines in your answer, or not, without affecting its correctness.

Each page with questions has the following instructions:

> ***Language*** **question:** The code on this page is used for questions x - y. If something is printed on a line that is a question (has a Qx comment, where "x" is a natural number) say what is printed. If the line has an error at either compile or runtime, answer "Err" and assume the statement doesn't exist for the rest of the program. If the statement prints nothing but is correct, answer "Ok". If a value is uninitialized, answer "uninit" or 0.

where "x" and "y" are question numbers, so you don't need to read these if you are running short on time. Page 6 has slightly different instructions reflecting the fact tht the question spans two pages.

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.

Name (must be signed to be graded):

Name

C++ question. The code on this page is used for questions 1 - 4. If something is printed on a line that is a question (has a Qx comment, where "x" is a natural number) say what is printed. If the line has an error at either compile or runtime, answer "Err" and assume the statement doesn't exist for the rest of the program. If the statement prints nothing but is correct, answer "Ok". If a value is uninitialized, answer "uninit" or 0.

```
// Box.h
template <typename T> class Box {
   T _data;
public:
   Box(const T& data);
   Box(const Box<T>& b);
   virtual ~Box( );
   T& getData( );
};

template <typename T> Box<T>::Box(
   const T& data) : _data(data)
{
   std::cout << data << std::endl;
}

template <typename T> Box<T>::Box(
   const Box<T>& b) : _data(b._data)
{
   std::cout << "c" << std::endl;
}

template <typename T> Box<T>::~Box( ) { }

template <typename T> T& Box<T>::getData( ) {
   return _data;
}

// Boxed.h
class Boxed {
   int key;
   int data;
public:
   Boxed(int key, int data);
   Boxed( );
   virtual ~Boxed( );
   friend ostream& operator<<(
          ostream& ostr, const Boxed& b);
};
```

```
// Boxed.cpp
Boxed::Boxed(int key, int data) :
   key(key), data(data) { }

Boxed::Boxed( ) : key(0), data(-1) { }

Boxed::~Boxed( ) { }

ostream& operator<< (
   ostream& os, const Boxed& m)
{
   os << "(" << m.key << ", " << m.data << ") ";
   return os;
}

// main.cpp
int main(void) {

   Box<int>* x1 = new Box<int>(9); // Q1
   Box<Boxed> x2(Boxed(1,2)); // Q2
   Box<Boxed> x3(x2); // Q3
   Box<Boxed> x4(x1); // Q4
}
```

C++ question. The code on this page is used for questions 5 - 7. If something is printed on a line that is a question (has a Qx comment, where "x" is a natural number) say what is printed. If the line has an error at either compile or runtime, answer "Err" and assume the statement doesn't exist for the rest of the program. If the statement prints nothing but is correct, answer "Ok". If a value is uninitialized, answer "uninit" or 0.

```
// A.h
class A {
public:
   int i;
   A(int d);
};

// A.cpp
A::A(int d) : i(d) {
   std::cout << "A" << std::endl;
}

// B.h
class B : public virtual A {
public:
   B(int d);
};

// B.cpp
B::B(int d) : A(d) {
   std::cout << "b" << std::endl;
}

// C.h
class C : public virtual A {
public:
   C(int d);
};

// C.cpp
C::C(int d) : A(d) {std::cout << "c" << std::endl;}
```

```
// D.h
class D : public B, C {
public:
   D(int d);
   void printD( );
};

// D.cpp
D::D(int d) : A(d), B(d-1), C(d-2) {
   std::cout << "d" << std::endl;
}

void D::printD( ) {
   // set the i accessible by the C
   // object to 99;
   C::i = 99;
   std::cout << C::i << ", " << B::i;
}

// main.cpp
int main(void) {

   D d(10); // Q5
   C c(5); // Q6

   d.i = 100;
   d.printD( ); // Q7
}
```

C++ question. The code on this page is used for questions 8 - 20. If something is printed on a line that is a question (has a Qx comment, where "x" is a natural number) say what is printed. If the line has an error at either compile or runtime, answer "Err" and assume the statement doesn't exist for the rest of the program. If the statement prints nothing but is correct, answer "Ok". If a value is uninitialized, answer "uninit" or 0.

```
// A.h
class A {
public:
   int i;
   int j;
   A( );
   A(A&);
   virtual ~A( );

   void f( );
   virtual void g( );
};

// A.cpp
A::A( ) {
   i = 1;
   j = 2;
   std::cout << "A";
}

A::A(A& a) {
   i = a.i;
   j = -a.j;
}

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

void A::f( ) {std::cout << "A::f";}

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

```
// B.h
class B : public A {
public:
   B( );
   virtual ~B( );

   void f( );
   void g( );
};

// B.cpp
B::B( ) {std::cout << "B";}
B::~B( ) {std::cout << "~B";}

void B::f( ) {std::cout << "B::f";}
void B::g( ) {std::cout << "B::g";}

// C.h
class C : protected A {
public:
   int m;
   int n;
   C(int);
};

// C.cpp
C::C(int k) : A( ), n(k), m(n) { };
```

```
// main.cpp
void foo(A& a) {
   a.i = 100;
}

void foo2(A a) {
   a.i = 99;
}

int main(void) {

   A a;
   A *aP = new B( ); // Q8
   C c(1);
   B b;
   std::cout << c.m << " " << c.n; // Q9
   std::cout << c.i; // Q10


   foo(a);
   std::cout << a.i; // Q11
   std::cout << a.j; // Q12

   foo2(a);
   std::cout << a.i; // Q13
   std::cout << a.j; // Q14

   aP->f( ); // Q15
   aP->g( ); // Q16

   A& aR = b;
   aR.f( ); // Q17
   aR.g( ); // Q18

   c.f( ); // Q19

   delete aP; // Q20
}
```

Java question. The code on this page is used for questions 21 - 24. If something is printed on a line that is a question (has a Qx comment, where "x" is a natural number) say what is printed. If the line has an error at either compile or runtime, answer "Err" and assume the statement doesn't exist for the rest of the program. If the statement prints nothing but is correct, answer "Ok". If a value is uninitialized, answer "uninit" or 0.

```
public class Obj {
   int val = 0;
}

class C1 implements Cloneable {
   public Obj obj=null;
   C1( ) {
      obj = new Obj();
      obj.val = 0;
   }

   C1(C1 c) {
      obj = new Obj();
      obj.val = 2;
   }
}

class C2 implements Cloneable {
   public Obj obj=null;
   C2( ) {
      obj = new Obj( );
      obj.val = 2;
    }

   public Object clone( ) throws
      CloneNotSupportedException
   {
      C2 rc1 = (C2) super.clone( );
      rc1.obj.val = rc1.obj.val+10;
      return rc1;
   }
}

class C3 implements Cloneable {
   public Obj obj=null;
   C3( ) {
      obj = new Obj( );
      obj.val = 3;
   }

   public Object clone( ) throws
      CloneNotSupportedException
   {
      C3 rc1 = (C3) super.clone( );
      return rc1;
   }
}
```

```
class Main {

   public static void main(String args[]) throws
      Exception
   {

      C1 c1 = new C1( );  // S1
      C2 c2 = new C2( );
      C3 c3 = new C3( );

      C1 c1Copy1 = new C1(c1);
      C1 c1Copy2 = c1;
      c1Copy2.obj.val = 99;
      System.out.println(""+c1.obj.val+" "+    // Q21
                         c1Copy1.obj.val+" "+  // Q21
                         c1Copy2.obj.val);     // Q21
      c1 = null; // S2

      C2 c2Copy = (C2) c2.clone( );
      System.out.println(""+c2.obj.val+" "+  // Q22
                         c2Copy.obj.val);    // Q22

      C3 c3Copy = (C3) c3.clone( );
      System.out.println(""+c3.obj.val+" "+ // Q23
                         c3Copy.obj.val);   // Q23

      c2Copy.obj.val = 100;
      c3Copy.obj.val = 100;
      System.out.println(""+c2.obj.val+" "+ // Q24
                         c3.obj.val);       // Q24
   }
}
```

Question 25: After statement S2 executes, is the object allocated in line S1 (the declaration of c1) garbage?

Java question. The code on this page is used for questions 26 - 28. If something is printed on a line that is a question (has a Qx comment, where "x" is a natural number) say what is printed. If the line has an error at either compile or runtime, answer "Err" and assume the statement doesn't exist for the rest of the program. If the statement prints nothing but is correct, answer "Ok". If a value is uninitialized, answer "uninit" or 0.

```
public class TestException1
   extends Exception {

   String str = "TE1";

   public TestException1( ) { }

   public String toString( ) {return str;}
}

public class TestException2
   extends Exception {

   String str = "TE2";

   public TestException2( ) { }

   public String toString( ) {return str;}
}
```

```
class Main {

   public static void f1(int i)
      throws TestException1
   {
      if (i < 1) {
         throw new TestException1( );
      }
   }

   public static void f2(int i)
      throws TestException2
   {
      if (i < 1) {
         throw new TestException2( );
      }
   }

   public static void main(String args[]) {
      for (int i = 0; i < 2; i++) {
         try {
            f1(i);
         } catch (TestException1 e) {
            System.out.println("caught "+e);
         } finally {
            System.out.println("F1 "+i);
         }
      }

      try {
         for (int j = 0; j < 2; j++) {
            f2(j);
         }
      } catch (TestException2 e) {
         System.out.println("caught "+e);
      } finally {
         System.out.println("F2 ");
      }
   }
}
```

Question 26. How many iterations of the i for loop will execute?
Question 27. How many iterations of the j for loop will execute?
Question 28. What is printed during the iteration i=0 of the i for loop?

Java question. The code on this page is used for questions 29 - 31. If something is printed on a line that is a question (has a Qx comment, where "x" is a natural number) say what is printed. If the line has an error at either compile or runtime, answer "Err" and assume the statement doesn't exist for the rest of the program. If the statement prints nothing but is correct, answer "Ok". If a value is uninitialized, answer "uninit" or 0.

```
public class Obj {
   int val = 0;
}
class C1 {
   public Obj obj=null;
   C1( ) {
      obj = new Obj();
      obj.val = 0;
   }

   C1(C1 c) {
      obj = new Obj();
      obj.val = 2;
   }
}
public class T implements Runnable {

   C1 c = null;

   public T(C1 cc) {
      c = cc;
   }

   public void run( ) {
      synchronized(c) {
         c.obj.val = 1;
      }
   }
}
```

```
class Main {

   public static void main(String args[])
      throws Exception
   {
      C1 c = new C1( );
      Thread t1 = new Thread(new T(c));
      Thread t2 = new Thread(new T(c));
      Thread t3 = new Thread(new T(new C1( )));
      Thread t4 = new Thread(new T(new C1( )));

      t1.run( ); // S1
      t2.run( ); // S2
      t1.start( );  // S3
      t2.start( );  // S4
      t1.join( ); t2.join( );

      t3.start( ); // S5
      t4.start( ); // S6
      t3.join( ); t4.join( );
   }
}
```

Question 29. Is there a race on the accesses to obj.val in the run methods of t1 and t2 in statement S1 and S2?

Question 30. Is there a race on the accesses to obj.val in the run methods of t1 and t2 in statement S3 and S4?

Question 31. Is there a race on the accesses to obj.val in the run methods of t3 and t4 in statement S5 and S6?

Java question. The code on this page is used for questions 32 - 39. If something is printed on a line that is a question (has a Qx comment, where "x" is a natural number) say what is printed. If the line has an error at either compile or runtime, answer "Err" and assume the statement doesn't exist for the rest of the program. If the statement prints nothing but is correct, answer "Ok". If a value is uninitialized, answer "uninit" or 0.

```
public interface I {
   public void f( );
}
public class B implements I {

   public static int stat = 1;

   public B( ) {stat++;}

   public void f( ) {
      System.out.println("B.f");
   }


   private void g1( ) {
      System.out.println("B.g1");
   }

   public void callG1( ) {g1( );}
}
public class C extends B {

   public void f( ) {
      System.out.println("C.f");
   }

   private void g1( ) {
      System.out.println("C.g1");
   }

   public void g2( ) {
      System.out.println("C.g2");
   }

}
```

```
class Main {

   public static void main(String args[])
   throws Exception
   {

      I i = new I( ); // Q32
      I b = new B( ); // Q33
      B b1 = new B( );
      B b2 = new B( );

      System.out.println(b1.stat); // Q34

      b.g1( ); // Q35

      b2 = new C( ); // Q36

      b2.f( ); // Q37

      b2.callG1( ); // Q38
      b2.g2( ); // Q39
   }
}
```