

**ECE 30862 Fall 2019 First Exam Answer Sheet**

Put your name and test version somewhere on this sheet. It's worth 5 points.

- |     |     |
|-----|-----|
| 1.  | 21. |
| 2.  | 22. |
| 3.  | 23. |
| 4.  | 24. |
| 5.  | 25. |
| 6.  | 26. |
| 7.  | 27. |
| 8.  | 28. |
| 9.  | 29. |
| 10. | 30. |
| 11. | 31. |
| 12. | 32. |
| 13. | 33. |
| 14. | 34. |
| 15. |     |
| 16. |     |
| 17. |     |
| 18. |     |
| 19. |     |
| 20. |     |

**This page intentionally left almost blank**

## ECE 39595J Fall 2019, Test 1

The first page is the answer sheet. You can annotate it and turn it in, you can turn in the entire exam with the answer sheet annotated, you can answer on a piece of scratch paper and turn in a .pdf (preferably) or as a .jpg.

You may begin the exam whenever it becomes available. I will give a 10 minute warning – at the end of that the exam answer sheet needs to have already been uploaded to Brightspace.

*If you are not in zoom with video turned on you may receive a 0 on the exam. I will be recording the exam. Check the Zoom chat box periodically for corrections.*

Programs are be given without “#include” statements for brevity. Assume all needed includes are present. “std::endl” may left off for brevity. You may use newlines in your answer, or not, without affecting your score.

Each question is worth 2.5 points except for the last question which is worth 10 points. You will receive 5 points for putting your name and test version on the answer sheet.

By taking and turning in a test answer sheet to be graded, you agree that: I have neither given nor received help during this exam from any other person or electronic source (other than my own notes, viewing the exam and using an electronic device to annotate the answer sheet with my answers), and I understand that if I have I will be guilty of cheating and will fail the exam and possibly the course.

This program has questions 1 through 26. If something is printed by a line that is a question (has a Qx comment, where “x” is a 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 when answering other questions. If the statement prints nothing but is correct, answer ”Ok”. If a printed value is uninitialized, answer ”uninit”.

```

public class Base {

    public int i = 0;

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

    public Base(int i) {
        System.out.println("Bi");
    }

    public void m0( ) {
        System.out.println("Bm0");
    }

    public void m1( ) {
        System.out.println("Bm1");
    }

    public void m2( ) {
        System.out.println("Bm2");
        m4( );
    }

    public void m3( ) {
        System.out.println("Bm3");
        m5( );
    }

    private void m4( ) {
        System.out.println("Bm4");
    }

    private void m5( ) {
        System.out.println("Bm5");
    }
}

public class Derived extends Base {

    public int i = 1;
    public int j = 1;

    public Derived( ) {
        super(1);
        System.out.println("D");
    }

    public Derived(int i) {
        System.out.println("Di");
    }

    public void m1( ) {
        System.out.println("Dm1");
    }

    public void m3( ) {
        System.out.println("Dm3");
        m5( );
    }

    private void m5( ) {
        System.out.println("Dm5");
    }

    public void m6( ) {
        System.out.println("Dm6");
    }
}

void B::a6( ) {std::cout << "B::a6 ";};

```

```
public class Main {  
  
    public static void main(String[ ] args) {  
  
        Base b = new Base( ); // Q1  
        b.m0( ); // Q2  
        b.m2( ); // Q3  
        b.m4( ); // Q4  
        b.m5( ); // Q5  
  
        Derived d = new Derived(1); // Q6  
  
        d.m0( ); // Q7  
        d.m1( ); // Q8  
        d.m2( ); // Q9  
        d.m3( ); // Q10  
        d.m4( ); // Q11  
        d.m5( ); // Q12  
        d.m6( ); // Q13  
  
        b = d; // Q14  
        b.m0( ); // Q15  
        b.m1( ); // Q16  
        b.m2( ); // Q17  
        b.m3( ); // Q18  
        b.m4( ); // Q19  
        b.m5( ); // Q20  
        b.m6( ); // Q21  
  
        d = b; // Q22  
  
        b = d; // Q23  
        System.out.println(b.i); // Q24  
        System.out.println(b.j); // Q25  
        System.out.println(d.i); // Q26  
    }  
}
```

This program has questions 27 - 29. If something is printed by 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 when answering other questions. If the statement prints nothing but is correct, answer ”Ok”. If a value is uninitialized, answer ”uninit”.

```

public class Int {
    private int value;

    public Int(int i) {
        value = i;
    }

    public void setValue(int i) {
        value = i;
    }

    public int getValue( ) {
        return value;
    }

    public String toString( ) {
        return ""+value;
    }
}

public class Main {

    private void fooI(int i) {
        i = -1;
    }

    private void fooInt(Int i, boolean change) {
        i.setValue(-i.getValue( ));
        if (change) i = new Int(7);
    }

    public static void main(String[ ] args) {

        Main m = new Main( );
        Int ii = new Int(1);
        int i = 2;
        double d = 2.;

        m.fooI(i);
        System.out.println("i: "+i); // Q27

        m.fooInt(ii, false);
        System.out.println("ii: "+ii); // Q28

        m.fooInt(ii, true);
        System.out.println("ii: "+ii); // Q29
    }
}

```

This program has questions 30 - 34. If something is printed by 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 when answering other questions. If the statement prints nothing but is correct, answer ”Ok”. If a value is uninitialized, answer ”uninit”.

```
public class Main {

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

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

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

    void foo(int i, float f) {
        System.out.println("if");
    }

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

    void foo(char c) {
        System.out.println("c");
    }

    public static void main(String[ ] args) {

        double d = 0.;
        float f = 0.f;
        long l = 0;
        int i = 0;
        short s = 0;
        char c = 0;
        Main m = new Main( );

        m.foo(i, f); // Q30
        m.foo(s, s); // 31
        m.foo(f, c); // 32
        m.foo(s); // Q33
        // Q34 What is the set of functions m.foo(i, f);
        // matches, i.e., the M set in the lectures

    }
}
```

Draw the VFT (virtual function table) for the two classes below. You can draw it on your answer sheet. If turning in a scan, and you use two sheets, either combine them into a single .pdf document or put them in a directory named your userid, zip it, and turn that in. This is worth 10 points.

```
public class B {  
  
    public void m0( ) {  
        System.out.println("m0");  
    }  
  
    public void m1( ) {  
        System.out.println("m1");  
    }  
  
    private void m2( ) {  
        System.out.println("m2");  
    }  
}  
  
public class D extends B {  
  
    public void m1( ) {  
        System.out.println("m1");  
    }  
  
    public void m4( ) {  
        System.out.println("m4");  
    }  
  
    private void m3( ) {  
        System.out.println("m3");  
    }  
}
```