ECE 30862 Fall 2018, Test 1

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. The total number of points should be 100, 97.5 from questions, 2.5 from signing both the test and the answer sheet. Each of the 39 questions is worth 2.5 points. After taking the test turn in both the test and the answer sheet.

Your exam should have 7 (seven) pages total (including this cover page and the answer sheet, one almost entire blank page 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. *Check the front board occasionally for corrections.*

Programs may be given without "#include" statements, and without "std::" for brevity, and to allow them to fit on a page. Assume these are present where needed.

For questions that are in comments at the ends of lines, e.g., "foo( ); // Q23", you should answer what is printed if something is printed, if nothing is printed answer and the statement is legal at both compile and runtime answer "Ok", and if nothing is printed by the statement gives either a compile time or run time error, answer "Error", "Err" or something similar. *If the statement is an error, answer questions on following lines in the program as if the statement did not exist in the program.*

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 (printed, worth 1 pt):

Last four digits of your ID:

The code below is used for questions 1 - 4.

```
// X1.h
#ifndef X1_H_
#define X1_H_

class X1 {
public:

   X1( );
   virtual ~X1( );
   static int i;
   int j;
};
#endif /* X1_H_ */
```

```
// X1.cpp
#include <iostream>
#include "X1.h"

X1::X1( ) { }
X1::~X1( ) { }

// main.cpp
#include <iostream>
#include "X1.h"
#include "D.h"

int main (int argc, char *argv[]) {
   X1 x( );
}
```

Q1: Pick the most correct statement.

a) X1 is the name of a class

b) X1 is the name of an object

c) X1 is the name of a variable

Q2: Pick the most correct statement.

a) x is the name of a class

b) x is the name of an object

c) x is the name of a variable

d) x is the name of an instance of a class

e) (a) and (b)

f) (a) and (c)

g) (b) and (c)

h) (b), (c) and (d)

Q3: If 10 objects of type X1 are created, how many copies of the variable i are there?

a) 0

b) 1

c) 10

Q4: If 10 objects of type X1 are created, how many copies of j are there?

a) 0

b) 1

c) 10

The code below is used for questions 5 - 24. For each line with a comment “Qx”, where ”x” is a natural number, say what is printed. If nothing is printed and the statement will cause an error, answer “Err”. If nothing is printed and the statement is legal, answer “Ok”.

```
// Person.h
class Person {
public:

   Person( );
   virtual ~Person( );
   virtual void getAddress( );
   virtual void getName( );
   void getID( );
   static void personCnt( );
   int i, j;
};

// Person.cpp
Person::Person( ) {i=0; j = 0; cout << "cP" << endl;}
Person::~Person( ) {cout << "dP" << endl;}
void Person::getAddress( ) {cout << "pA" << endl;}
void Person::getName( ) {cout << "pN" << endl;}
void Person::getID( ) {cout << "pI" << endl;}
void Person::personCnt( ) {cout << "pC" << endl;}

// Student.h
class Student : public Person {
public:

   Student( );
   virtual ~Student( );
   void getName( );
   virtual void getID( );
   virtual void tuition( );
   static void personCnt( );
   int j;
};

// Student.cpp
Student::Student( ) {j=1; cout << "cS" << endl;}
Student::~Student( ) {cout << "dS" << endl;}
void Student::getName( ) {cout << "sN" << endl;}
void Student::getID( ) {cout << "sI" << endl;}
void Student::tuition( ) {cout << "sT" << endl;}
void Student::personCnt( ) {cout << "sC" << endl;}
```

```
// GradStudent.h
class GradStudent : public Student {
public:

   GradStudent( );
   virtual ~GradStudent( );
   void getID( );
};

// GradStudent.cpp
GradStudent::GradStudent( ) {cout << "cG" << endl;}
GradStudent::~GradStudent( ) {cout << "dG" << endl;}
void GradStudent::getID( ) {cout << "gI" << endl;}
#include <iostream>
#include "Person.h"
#include "Student.h"
#include "GradStudent.h"

// main.cpp
int main (int argc, char *argv[]) {
   Person p;
   Student s; // Q5
   GradStudent g;
   Person* tp = new Person( );
   Student* ts = new Student( );
   GradStudent* tg = new GradStudent( );

   p = s;
   p.getAddress( ); // Q6
   p.getID( ); // Q7
   p.getName( ); // Q8

   tp->tuition( ); // Q9
   tp->getID( ); // Q10
   cout << tp->j << endl; // Q11
   tp->getName( ); // Q12

   tp = ts;
   tp->tuition( ); // Q13
   tp->getName( ); // Q14
   tp->getID( ); // Q15
   tp->personCnt( ); // Q16
   cout << tp->j << endl; // Q17

   ts = tg;
   ts->personCnt( ); // Q18
   ts->getID( ); // Q19
   ts->getName( ); // Q20
   cout << ts->i << endl; // Q21
   cout << ts->j << endl; // Q22

   s = p; // Q23
   ts = tp; // Q24
}
```

The code below is used for questions 25 - 33. For each line with a comment “Qx”, where ”x” is a natural number, say what is printed. If nothing is printed and the statement will cause an error, answer “Err”. If nothing is printed and the statement is legal, answer “Ok”.

```
// Base.h
class Base {
public:

   Base( );
   virtual ~Base( );
   virtual void f(int);
   virtual void f(float);
   static int sc;
   int i;
};

// Base.cpp
int Base::sc = 0;

Base::Base( ) {
   sc++;
   i = 0;
}

Base::~Base( ) {cout << "dB" << endl;}

void Base::f(int k) {cout << "bk" << endl;}
void Base::f(float f) {cout << "bf" << endl;}

// Derived.h
class Derived : public Base {
public:

   Derived( );
   virtual ~Derived( );
   void f(float);

};
```

```
// Derived.cpp
Derived::Derived( ) {sc++;}
Derived::~Derived( ) {cout << "dD" << endl;}

void Derived::f(float) {cout << "df" << endl;}

// main.cpp
void foo(Base b) {
   b.i++;
};

void foo(Base* tb) {
   tb->i++;
};

// main.cpp
int main (int argc, char *argv[]) {
   Base b;
   cout << Base::sc << endl; // Q25
   Derived d;
   cout << Base::sc << endl; // Q26

   Base* tb = &b;
   Derived* td = &d;

   foo(b); // Q27
   cout << b.i << endl; // Q28

   foo(tb); // Q29
   cout << tb->i << endl; // Q30

   tb = td;
   tb->f(2); // Q31
   td->f(2); // Q32
}
```

Q33: What is printed by the destructors when function **main** is exited?

a) dD dB dB

b) dD dB

c) dB dD dB

d) dB dB dD

e) None of the above

The code below is used for questions 33 - 36. For each line with a comment “Qx”, where ”x” is a natural number, say what is printed. If nothing is printed and the statement will cause an error, answer “Err”. If nothing is printed and the statement is legal, answer “Ok”.

```
// Base.h
class Base {
public:

   Base( );
   virtual ~Base( );
   virtual void f(int)=0;
};

// Base.cpp
Base::Base( ) { }

Base::~Base( ) { }
#ifndef D1_H_
#define D1_H_
#include "Base.h"

// D1.h
class D1 : public Base {
public:

   D1( );
   virtual ~D1( );
   virtual void g( );
};

// D1.cpp
D1::D1( ) { }
D1::~D1( ) { }

void D1::g( ) {cout << "dg" << endl;}
```

```
// D2.h
class D2 : public D1 {
public:

   D2( );
   virtual ~D2( );
   void f(int);
};

// D2.cpp
D2::D2( ) { }
D2::~D2( ) { }

void D2::f(int i) {cout << "df" << endl;}
#include <iostream>
#include "Base.h"
#include "D1.h"
#include "D2.h"

// main.cpp
int main (int argc, char *argv[]) {
   Base* tb1 = new Base( );
   D1* td1 = new D1( ); // Q34
   D2* td2 = new D2( ); // Q35
   Base* tb2 = td2;
   tb2->f(1); // Q36
}
```

The code below is used for questions 37 - 39. For each line with a comment “Qx”, where ”x” is a natural number, say what is printed. If nothing is printed and the statement will cause an error, answer “Err”. If nothing is printed and the statement is legal, answer “Ok”.

```
// Base.h
class Base {
public:

   Base( );
   virtual ~Base( );
};

// Base.cpp
Base::Base( ) { }

Base::~Base( ) { }

// D1.h
class D1 : public Base {
public:

   D1( );
   virtual ~D1( );
};

// D1.cpp
D1::D1( ) { }
D1::~D1( ) { }
```

```
// D2.h
class D2 : public Base {
public:

   D2( );
   virtual ~D2( );
};

// D2.cpp
D2::D2( ) { }
D2::~D2( ) { }

// main.cpp
int main (int argc, char *argv[]) {
   Base* tb1 = new D1( );
   Base* tb2 = new D2( );
   tb1 = tb2; // Q37

   D1* td1 = new D1( );
   D2* td2 = new D2( );

   td1 = static_cast<D1*>(td2); // Q38
   td1 = dynamic_cast<D1*>(td2);
   // only one Q39 line will execute.  Say
   // what is printed by that line.
   if (td1 == NULL)
      std::cout << "N" << std::endl; // Q39
   else
      std::cout << "!" << std::endl; // Q39
}
```

ECE 30862 Fall 2018 First Exam Answer Sheet

All answers should be on this sheet. Both this sheet and your test must be signed and turned in. You may detach this sheet from the rest of the test to make it easier to write your answers on it. Each question is worth 4 points.

I promise that I have neither Given nor received disallowed aid on this test.

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.