ECE 462 Exam 1

08:10-09:20AM, October 02, 2009

I certify that I will not receive nor provide aid to any other student for this exam.

Signature:

You must sign here. Otherwise, the exam is not graded.

This exam is printed double sides.

Write your answers next to the questions. If you need more space, you can use the two blank pages.

This is an open-book, open-note exam. You can use any book or note or program printouts.

Please turn off your cellular phone now.

Two outcomes are tested in this exam. To pass each outcome, you must receive 50% of the points.

• Outcome 3: an understanding of the concepts of inheritance and polymorphism.
• Outcome 4: an ability to use template classes and the STL library in C++ and Java.

You need to obtain 50% of the points in each outcome to pass the outcome. There are 12 questions: If a program has a syntax error, point out which line causes the error. If there are multiple errors, you need to write only one of them.

If a question has a numeric answer, you can write the procedure without the final result. For example, you can write “1 + 2” instead of “3”.

1
This page is blank.
This page is blank.
1 Inheritance and Class (Outcome 3, 0.5 point)

Which statement is correct? You may choose multiple answers.

Answer: [A]

A. An object in a derived class contains all attributes in a base class, including private attributes.
B. If a function is provided in a based class, the function must be overridden in a derived class.
C. In C++, if a class has no destructor, memory leak occurs.
D. In Java, a built-in class (such as Vector) cannot be extended.
E. If a class X extends another class Y, X cannot be further extended.

2 Inheritance and Functions (Outcome 3, 0.5 point)

Which statement is correct? You may choose multiple answers.

Answer: [D]

A. If a Java class has a virtual function, this class is abstract.
B. A derived class must override all functions in an abstract base class.
C. In Java, if a class has private functions, the class cannot have derived classes.
D. An object’s attributes may also be objects.
E. If a C++ class has a constructor, the class must have a destructor.

3 C++ Class Hierarchy (Outcome 3, 1 point)

Complete the program.

```cpp
#include <iostream>
using namespace std;

class Person
{
    public:
        Person(string n, string a)
        {
```
```cpp
name = n;
address = a;
}
virtual void introduce()
{
    cout << "My name is " << name << "." << endl;
    cout << "I live in " << address << "." << endl;
}
private:
    string name;
    string address;
);

class Student:public Person
{
public:
    Student(string n, string a, string s, string m)
    /*
    * ---> FIX ME <---
    * initialize name, address, school, and major
    * 
    */
    virtual void introduce()
    {
    /*
    * ---> FIX ME <---
    * print name, address, school, and major
    * 
    */
    }
private:
    string school;
    string major;
};

int main(int argc, char * argv[])
{
    Person p1("Amy", "West Lafayette");
    Student s2("David", "Lafayette", "Purdue", "ECE");
    p1.introduce();
    s2.introduce();
    p1 = s2;
    p1.introduce();
};
```
return 0;
}

#include <iostream>
using namespace std;
class Person
{
public:
Person(string n, string a)
{
    name = n;
    address = a;
}
virtual void introduce()
{
    cout << "My name is " << name << "." << endl;
    cout << "I live in " << address << "." << endl << endl;
}
private:
    string name;
    string address;
};
class Student:public Person
{
public:
Student(string n, string a, string s, string m): Person(n, a)
{
    /* cannot use
       name = n;
       address = a;
       because they are private */
    school = s;
    major = m;
}
void introduce()
{
    Person::introduce();
    cout << "I study at " << school << "." << endl;
    cout << "My major is " << major << "." << endl << endl;
}
private:
    string school;
int main(int argc, char * argv[]) {
    Person p1("Amy", "West Lafayette");
    Student s2("David", "Lafayette", "Purdue", "ECE");
    p1.introduce();
    s2.introduce();
    p1 = s2;
    p1.introduce();
    return 0;
}

4 Java Derived Class (Outcome 3, 1 point)

What is the value of p_val after executing the statements at /* A */, /* B */, ..., /* E */? Concisely explain your answers. Your answer should contain five numbers (0.1 point each) and an explanation (0.5 point).

Answer:

0 2 2 13 18

import java.io.*;
import java.util.*;

class Person {
    protected int p_val;
    public Person() { p_val = 0; }
    public void func1() { p_val += 1; }
    public void func2() { p_val += 2; }
    public int getVal() {
        func1();
        return p_val;
    }
}

class Student extends Person {
    public Student() { p_val += 3; }
    public void func1() { p_val += 4; }
}
class CollegeStudent extends Student
{
    public CollegeStudent() { p_val += 5; }
    public void func2() { p_val += 6; }
}

class question34
{
    public static void main(String[] args)
    {
        Person [] per = new Person[3];
        per[0] = new Person();
        per[1] = new Student();
        per[2] = new CollegeStudent();
        Student [] stu = new Student[2]; /* A */
        if ((per[1].getVal() % 2) == 1)
        {
            stu[0] = new Student();
        } else
        {
            stu[0] = new CollegeStudent();
        }
        if ((per[0].getVal() % 2) == 0)
        {
            stu[1] = new Student();
        } else
        {
            stu[1] = new CollegeStudent();
        }
        per[0].func1(); /* B */
        per[1].func2();
        stu[0].func1(); /* C */
        stu[1].func2();
        System.out.println(per[1].getVal()); /* D */
        System.out.println(stu[1].getVal()); /* E */
    }
}
5 C++ Derived Class and Static Attribute (Outcome 3, 1 point)

What is the value of `p_val` after executing the statements at /* A */ /* B */ ... /* E */? Concisely explain your answers. Your answer should contain five numbers (0.1 point) and an explanation (0.5 point).

Answer:

11 28 34 40 44

#include <iostream>
using namespace std;

class Person
{
public:
    Person() { ;
    /*** ATTENTION ***/
    virtual void func1() { p_val += 1; } / * ATTENTION: virtual */
    void func2() { p_val += 2; }
    int getVal()
    {
        func1();
        return p_val;
    }
protected:
    /*** ATTENTION ***/
    static int p_val; / *** ATTENTION: static ***/
};

int Person::p_val = 0;

class Student: public Person
{
    public:
    Student() { p_val += 3; }
    void func1() { p_val += 4; }
};

class CollegeStudent: public Student
{
    public:
    CollegeStudent() { p_val += 5; }
    void func2() { p_val += 6; }
};
int main(int argc, char * argv[]) {
    Person * per[3];
    per[0] = new Person();
    per[1] = new Student();
    per[2] = new CollegeStudent();
    Student * stu[2]; /* A */
    if ((per[1] -> getVal() % 2) == 1)
    {
        stu[0] = new Student();
    }
    else
    {
        stu[0] = new CollegeStudent();
    }
    if ((per[0] -> getVal() % 2) == 0)
    {
        stu[1] = new Student();
    }
    else
    {
        stu[1] = new CollegeStudent();
    }
    per[0] -> func1(); /* B */
    per[1] -> func2();
    stu[0] -> func1(); /* C */
    stu[1] -> func2();
    cout << per[1] -> getVal() << endl; /* D */
    cout << stu[1] -> getVal() << endl; /* E */
}

6 Java Set (Outcome 4, 1 point)

Use Java list to implement set. For simplicity, the set can handle strings only.

import java.io.*;
import java.util.*;

class ECE462Set
{
    public ECE462Set()
    {
        /* ---> FIX ME <--- */
    }
}
public void add(String s)
    /*
     * add a string to the set
     * must check duplicate first
     */
    {
        /* ---> FIX ME <--- */
    }

public void remove(String s)
    /*
     * remove s from the list
     * do nothing if s is not in the set
     */
    {
        /* ---> FIX ME <--- */
    }

public Iterator iterator()
    {
        /* ---> FIX ME <--- */
    }

public boolean contain(String s)
    {
        /* ---> FIX ME <--- */
    }

private ArrayList<String> slist;

class question41
{
    public static void main(String[] args)
    {
        ECE462Set eset = new ECE462Set();
eset.add("c++");
eset.add("fall");
eset.add("c++");
eset.add("2009");
eset.add("java");
eset.remove("java");
eset.remove("spring");
eset.add("2008");
System.out.println(eSet.contains("fall"));
System.out.println(eSet.contains("spring"));
Iterator iter = eSet.iterator();
while (iter.hasNext()) {
    System.out.println(iter.next());
}
}

Answer:
import java.io.*;
import java.util.*;

class ECE462Set {
    public ECE462Set() {
        sList = new ArrayList<String>();
    }
    public void add(String s) {
        /*
         * add a string to the set
         * must check duplicate first
         * /
        if (sList.contains(s)) {
            System.out.println("already contains " + s);
            return;
        }
        sList.add(s);
    }
    public void remove(String s) {
        /*
         * remove a string from the list
         * do nothing if s is not in the set
         * /
        if (sList.contains(s)) {
            System.out.println("remove " + s);
            sList.remove(s);
        }
    }
}
public Iterator iterator()
{
    return slist.iterator();
}

public boolean contain(String s)
{
    return slist.contains(s);
}

private ArrayList<String> slist;

class question41
{
    private static void printSet(ECE462Set es)
    {
        Iterator iter = es.iterator();
        while ( iter.hasNext() )
        {
            System.out.println( iter.next() );
        }
        System.out.println("");
    }
    public static void main(String[] args)
    {
        ECE462Set eset = new ECE462Set();
eset.add("c++");    printSet(eset);
eset.add("fall");    printSet(eset);
eset.add("c++");    printSet(eset);
eset.add("2009");   printSet(eset);
eset.add("java");   printSet(eset);
eset.remove("java"); printSet(eset);
eset.remove("spring"); printSet(eset);
eset.add("2008");   printSet(eset);
System.out.println(eset.contain("fall"));
System.out.println(eset.contain("spring"));
        Iterator iter = eset.iterator();
        while ( iter.hasNext() )
        {
            System.out.println( iter.next() );
        }
    }
}
7 Container Classes (Outcome 4, 0.5point)

Which statement is correct? You may choose multiple answers.

*Answer: B*

A. It is **not** possible to create a set using a map (or multiple maps) because a map requires (key, value) pairs.

B. An element can be inserted anywhere in a list.

C. In a map, both the values and the keys **must** be unique.

D. In Java, an array is an object; hence, Java automatically allocates more memory if an index exceeds the current size of the array.

E. In C++, a list of `int` is **not** allowed. In Java, a list of `int` is allowed.

8 C++ Queue (Outcome 4, 1 point)

Implement C++ queue using list or set. This queue can handle different types, including `int` and `string`; therefore, you should use template.

```cpp
#include <iostream>
#include <string>
#include <list>
using namespace std;

/* ---> FIX ME <--- */
class ECE462Queue
{
public:
    void add(TYPE v)
    /* add an element to the end of queue */
    {
        /* ---> FIX ME <--- */
    }
    TYPE remove()
    /* return and remove the first element from the beginning */
};
*/
TYPE front()
/* return the first element from the beginning without changing the queue */
{
    /* ---> FIX ME <--- */
}

int size()
/* return the number of elements in the queue */
{
    /* ---> FIX ME <--- */
}

private:
/* use a (or multiple) list or set */
/* ---> FIX ME <--- */

};

int main(int argc, char * argv[])
{
    ECE462Queue<int> iqueue;
    iqueue.add(4);
    iqueue.add(6);
    iqueue.add(2);
    cout << iqueue.remove() << endl;
    cout << iqueue.size() << endl;
    cout << iqueue.remove() << endl;
    cout << iqueue.remove() << endl;
    cout << iqueue.size() << endl; // endl
    cout << iqueue.size() << endl; // endl
    ECE462Queue<string> squeue;
    squeue.add("java");
    squeue.add("programming");
    squeue.add("fall");
    squeue.add("purdue");
    squeue.remove();
    cout << squeue.front() << endl;
    squeue.remove();
    cout << squeue.front() << endl;
    cout << squeue.front() << endl;
    cout << squeue.size() << endl;
    return 0;
}

Answer:
#include <iostream>

16
#include <string>
#include <list>
using namespace std;

template <class TYPE>
class ECE462Queue
{
    public:
        void add(TYPE v)
        /* add an element to the end of queue */
            { 
                values.push_back(v);
            }
    TYPE remove()
        /* return and remove the first element from the beginning */
        { 
            TYPE v = values.front();
            values.pop_front();
            return v;
        }
    TYPE front()
        /* return the first element from the beginning without changing the 
          queue */
        { 
            TYPE v = values.front();
            return v;
        }
    int size()
        { 
            return values.size();
        }
    private:
        list<TYPE> values;
};

int main(int argc, char * argv[])
{
    ECE462Queue<int> iqueue;
    iqueue.add(4);
    iqueue.add(6);
    iqueue.add(2);
    cout << iqueue.remove() << endl;
    cout << iqueue.size() << endl;
    cout << iqueue.remove() << endl;
    cout << iqueue.remove() << endl;
    cout << iqueue.remove() << endl;
cout << iqueue.size() << endl << endl;

ECE462Queue<string> squeue;
squeue.add("java");
squeue.add("programming");
squeue.add("fall");
squeue.add("purdue");
squeue.remove();
cout << squeue.front() << endl;
squeue.remove();
cout << squeue.front() << endl;
cout << squeue.size() << endl;
return 0;
}

9 Java Container (Outcome 4, 1 point)

What is the output of this program (0.5 point)? Explain the answers concisely (0.5 point).

Answer:
100 10 90 0 10

import java.util.*;

public class question44 {

    public static void main(String[] args) {
        List<Integer> list = new ArrayList<Integer>();
        Set<Integer> set = new HashSet<Integer>();
        Map<Integer, Integer> map = new TreeMap<Integer, Integer> ();

        Integer i, j;
        for(i = 0; i < 100; i++)
        {
            j = i % 10;    // modulo
            list.add(j);
            set.add(j);
            map.put(j, i);
        }

        System.out.println(list.size());
        System.out.println(map.size());
        System.out.println(map.get(0));
    }
}
10 Container Classes (Outcome 4, 0.5point)

Which statement is correct? You may choose multiple answers.

Answer: C

A. It is always better to use a list than a vector because new elements can be inserted into a list, not a vector.
B. It is always better to use a set than a map because elements in a set must be distinct.
C. If insertions are rare, a vector is faster than a list for random accesses.
D. If insertions are frequent, a queue is faster than a stack for random accesses.
E. C++ does not provide garbage collection. Hence, it is necessary to allocate sufficient memory before using a map.

11 Object-Oriented Design, 1 point

Please read the question carefully. Your answer should be more than “Yes” or “No.” Keep your answers concise. Use meaningful function names and do not provide the functions’ implementations.

In a Tetris game, should each piece be an object? List two reasons why it should or should not be an object.

If a piece should be an object, what functions should be provided as the class’ member functions?

If a piece should not be an object, what functions are related to a piece?

Answer: Yes. 1. encapsulate the representation 2. handle rotation 3. compare equal pieces
12 Debugging, 1 point

There are many ways debugging a program. You may use a debugger or insert printing state-
ments. List one scenario in which using an interactive debugger (such as gdb) is better than
printing. List one scenario in which printing is better than using a debugger. Please keep your
answers concise.

Answer:

gdb better than printf: no need to recompile, breakpoint, conditional breakpoint, step statements, dereference
pointers, back trace to see call stack (very useful when a program encounters segmentation fault)

printf better than gdb: timing sensitive, long-running program, logging for post-execution analysis, opti-
mized code (using -O)

wrong answer: “syntax error” by compiler, neither by gdb not by printf