**1** Which statements about the hashCode() and equals() methods are true?

Select the **two** correct answers.

(a) Two objects that are different according to the equals() method, must have

different hash values.

(b) Two objects that are equal according to the equals() method, must have the

same hash value.

(c) Two objects that have the same hash value, must be equal according to the

equals() method.

(d) Two objects that have different hash values, must be unequal according to the

equals() method.

Ans : b,d

**2** Given that the objects referenced by the parameters override the equals() and the hashCode() methods appropriately, which return values are possible from the following method?

String func(Object x, Object y) {

return (x == y) + " " + x.equals(y) + " " + (x.hashCode() == y.hashCode());

}

Select the **four** correct answers.

(a) "false false false"

(b) "false false true"

(c) "false true false"

(d) "false true true"

(e) "true false false"

(f) "true false true"

(g) "true true false"

(h) "true true true"

Ans : a,b,d,h

**3** Which code, when inserted at (1), in the equalsImpl() method will provide a correct implementation of the equals() method?

public class Pair {

int a, b;

public Pair(int a, int b) {

this.a = a;

this.b = b;

}

public boolean equals(Object o) {

return (this == o) || (o instanceof Pair) && equalsImpl((Pair) o);

}

private boolean equalsImpl(Pair o) {

// (1) INSERT CODE HERE ...

}

}

Select the **three** correct answers.

(a) return a == o.a || b == o.b;

(b) return false;

(c) return a >= o.a;

(d) return a == o.a;

(e) return a == o.a && b == o.b;

Ans : b, d, e

**4** Which code, when inserted at (1), will provide a correct implementation of the

hashCode() method in the following program?

import java.util.\*;

public class Measurement {

int count;

int accumulated;

public Measurement() {}

public void record(int v) {

count++;

accumulated += v;

}

public int average() {

return accumulated/count;

}

public boolean equals(Object other) {

if (this == other)

return true;

if (!(other instanceof Measurement))

return false;

Measurement o = (Measurement) other;

if (count != 0 && o.count != 0)

return average() == o.average();

return count == o.count;

}

public int hashCode() { // (1) INSERT CODE HERE ... } }

Select the **two** correct answers.

(a) return 31337;

(b) return accumulated / count;

(c) return (count << 16) ^ accumulated;

(d) return ~accumulated;

(e) return count == 0 ? 0 : average();

Ans : a, e

**5** What will be the result of compiling and running the following program?

import java.util.Comparator;

class Person implements Comparable<Person> {

String name;

int age;

Person(String name, int age)

{ this.name = name; this.age = age; }

public int compareTo(Person p2) {

Comparator<String> strCmp = Person.cmp();

int status = strCmp.compare(this.name, p2.name);

if (status == 0) {

Comparator<Integer> intCmp = Person.cmp();

status = intCmp.compare(this.age, p2.age);

}

return status;

}

public static <E extends Comparable<E>> Comparator<E> cmp() {

return new Comparator<E>() {

public int compare(E s1, E s2) { return s2.compareTo(s1); }

};

}

public static void main(String[] args) {

Person p1 = new Person("Tom", 20);

Person p2 = new Person("Dick", 30);

Person p3 = new Person("Tom", 40);

System.out.println((p1.compareTo(p2) < 0) + " " + (p1.compareTo(p3) < 0));

}

}

Select the one correct answer.

(a) The program will fail to compile.

(b) The program will compile but throw an exception when run.

(c) The program will compile and print true false, when run.

(d) The program will compile and print true true, when run.

(e) The program will compile and print false false, when run.

(f) The program will compile and print false true, when run.

Ans : c

**6** Which of these are core interfaces in the collections framework?

Select the three correct answers.

(a) Set<E>

(b) Bag<E>

(c) LinkedList<E>

(d) Collection<E>

(e) Map<K,V>

Ans : a,d,e

**7** Which of these implementations are provided by the java.util package?

Select the two correct answers.

(a) HashList<E>

(b) HashMap<K,V>

(c) ArraySet<E>

(d) ArrayMap<K,V>

(e) TreeMap<K,V>

Ans : b,e

**8** What is the name of the interface used to represent collections that maintain non-unique

Elements in order?

Select the one correct answer.

(a) Collection<E>

(b) Set<E>

(c) SortedSet<E>

(d) List<E>

(e) Sequence<E>

Ans : d

**9** Which methods are specified by the Iterator<E> interface?

Select the three correct answers.

(a) hasNext()

(b) hasMore()

(c) remove()

(d) delete()

(e) more()

(f) next()

Ans : a, c, f

**10** Which identifiers, when inserted in appropriate places in the program, will result

in the output 911?

Collection<\_\_\_\_\_\_\_\_> myItems = new ArrayList<\_\_\_\_\_\_\_\_\_\_>();

myItems.add(9); myItems.add(1); myItems.add(1);

Iterator<\_\_\_\_\_\_\_\_\_> iterator = \_\_\_\_\_\_\_\_\_\_\_\_\_.iterator();

while (\_\_\_\_\_\_\_\_\_\_\_\_\_\_.\_\_\_\_\_\_\_\_\_\_\_()) {

System.out.print(\_\_\_\_\_\_\_\_\_\_\_\_\_.\_\_\_\_\_\_\_\_\_());

}

Select the five correct answers.

(a) hasNext

(b) myItems

(c) next

(d) Integer

(e) int

(f) Collection

(g) iterator

Ans : d,b,g,a,c