

Questions #3 and #4 refer to the following interface and class definition:

```
public interface Container
{
    public void insert(Object x);    // insert x into Container
    public Object remove( );        // remove item from Container
}
public class C implements Container
{
    public C( ) // constructor
    { implementation code }
    public void insert(Object x)    // insert x into C
    { implementation code }
    public Object remove( )        // remove item from C
    { implementation code }
    //appropriate private instance variables to implement C
}
```

Here is a program segment that uses class C above:

```
Container words = new C( );
String w1 = "Dog";
String w2 = "Cat";
String w3 = "Pig";
String w4 = "Bird";
words.insert(w1);
words.insert(w2);
words.insert(w3);
words.insert(w4);
String str;
str = (String) words.remove( );
str = (String) words.remove( );
System.out.println(str);
```

3. What will the output be if C is a stack? 3) _____

4. What will the output be if C is a queue? 4) _____

I. Answer the following (2 pts each)

- 1) Methods `add(s)` and `multiply(s)` do the following to a stack `s` when invoked:

The stack is popped twice.

The two popped items are added or multiplied accordingly.

The Integer result is pushed onto the stack.

What will stack `s` contain following execution of the following code segment?

```
int x = 2, y = 6, a = 10, b = 8;
```

```
s.push(new Integer(x));
```

```
s.push(new Integer(y));
```

```
add(s); // pops 2, adds, and pushes result
```

```
s.push(new Integer(a));
```

```
s.push(new Integer(b));
```

```
multiply(s); // pops 2, multiplies, and pushes result
```

```
add(s); // pops 2, adds, and pushes result
```

1) _____

- 2) Assume these declarations:

```
Queue q = new ListQueue( ); // ListQueue implements Queue
```

```
int sum = 0;
```

```
Object obj;
```

```
Integer a = new Integer(1);
```

```
Integer b = new Integer(9);
```

```
Integer c = new Integer(5);
```

```
q.enqueue(a);
```

```
q.enqueue(b);
```

```
q.enqueue(c);
```

```
a = new Integer(a.intValue() + 1);
```

```
a = (Integer) q.dequeue( );
```

```
q.enqueue(c);
```

```
q.enqueue(a);
```

```
b = a;
```

```
q.enqueue(b);
```

```
while (!q.isEmpty( ))
```

```
{
```

```
    a = (Integer) q.dequeue( );
```

```
    System.out.print(a);
```

```
}
```

What output will be produced?

2) _____

Over