

CSC 1351 The Twelve Hour Exam From Hell

Name: _____

1 Arrays (Ch. 6)

1.1

Does this code compile? If it does, what is its output?

```
import java.util.*;
public class L {
    int[] data;
    void append(int n) {
        int[] newData = new int[data.length+1];
        for(int i=0;i<data.length;i++)
            newData[i] = data[i];
        newData[data.length] = n;
        data = newData; }
    public static void main(String[] args) {
        L l = new L();
        l.append(3);
        System.out.println(Arrays.toString(l.data)); } }
```

NullPointerException

1.2

Does this code compile? If it does, what is its output?

```
import java.util.*;
public class L {
    int[] data = new int[1];
    void append(int n) {
        int[] newData = new int[data.length+1];
        for(int i=0;i<data.length;i++)
            newData[i] = data[i];
        newData[data.length] = n;
        data = newData; }
    public static void main(String[] args) {
        L l = new L();
        l.append(3);
        System.out.println(Arrays.toString(l.data)); } }
```

[0, 3]

1.3

Does this code compile? If it does, what is its output?

```
import java.util.*;
public class L {
    int[] data = new int[0];
    void append(int n) {
        int[] newData = new int[data.length+10];
        for(int i=0;i<data.length;i++)
            newData[i] = data[i];
        newData[data.length] = n;
        data = newData; }
```

```

public static void main(String[] args) {
    L l = new L();
    l.append(3);
    System.out.println(Arrays.toString(l.data)); } }

```

[3, 0, 0, 0, 0, 0, 0, 0, 0, 0]

1.4

Does this code compile? If it does, what is its output?

```

public class Sum {
    int[] data = new int []{1,5,3,9,2,4};
    public static void main(String[] args) {
        int sum = 0;
        for(int i=0;i<data.length-1;i++)
            sum += data[i];
        System.out.println("sum="+sum);
    }
}

```

compile error:

```

Sum.java:6: error: non-static variable data cannot be referenced from a static context
    for(int i=0;i<data.length-1;i++)
        ^

```

```

Sum.java:7: error: non-static variable data cannot be referenced from a static context
    sum += data[i];
        ^

```

2 errors

1.5

Does this code compile? If it does, what is its output?

```

public class Sum {
    public static void main(String[] args) {
        int[] data = new int []{1,5,3,9,2,4};
        int sum = 0;
        for(int i=0;i<data.length-1;i++)
            sum += data[i];
        System.out.println("sum="+sum);
    }
}

```

sum=20

1.6

What could go wrong with this code? Compute all primes.

```

static List<Integer> primesLessThanN(int n) {
    List<Integer> primes = new ArrayList<>();
    primes.add(2);
    for(int i=3;i<n;i+=2) {
        boolean isPrime = true;
        for(Integer p : primes) {
            if(i % p == 0) {
                isPrime = false;
                break;
            }
        }
        if(isPrime)

```

```

    primes.add(i);
}
return primes;
}

```

It does not work for $n \leq 2$

1.7

Write a program to compute an List of Integers containing all the digits in an integer N.

```

static List<Integer> digits(int N) {
    List<Integer> d = new ArrayList<>();
    while(true) {
        d.add(N % 10);
        N /= 10;
        if(N == 0) break;
    }
    return d;
}

```

1.8

Rerite the above program using arrays.

```

static int[] digits(int N) {
    int M = N; int size = 0;
    while(true) {
        size++;
        N /= 10;
        if(N == 0) break;
    }
    N = M; int pos = 0;
    int[] res = new int[size];
    while(true) {
        res[pos++] = N % 10;
        N /= 10;
        if(N == 0) break;
    }
    return res;
}

```

1.9

Does this code compile? If so what is its output?

```

import java.util.*;
public class Buf {
    int[] data;
    public Buf(int n) { data = new int[n]; }
    int pos = 0;
    public void push(int v) { data[pos++ % data.length] = v; }
    public String toString() { return Arrays.toString(data); }
    public static void main(String[] args) {
        Buf b = new Buf(3);
        for(int i=0;i<10;i++) b.push(i);
        System.out.println("b="+b);
    }
}

```

$b=[9, 7, 8]$

1.10

Does this code compile? If so what is its output?

```
import java.util.*;
public class Sq {
    public static void main(String[] args) {
        List<Integer> li = new ArrayList<>();
        for(int i=0;i<3;i++)
            for(int j=0;j<=i;j++)
                li.add(j+1);
        System.out.println(li);
    }
}
```

[1, 1, 2, 1, 2, 3]

1.11

Does this code compile? If so what is its output?

```
import java.util.*;
public class Sq {
    public static void main(String[] args) {
        List<Integer> li = new ArrayList<>();
        for(int i=0;i<3;i++)
            for(int j=i;j>=0;j--)
                li.add(j*j);
        System.out.println(li);
    }
}
```

[0, 1, 0, 4, 1, 0]

1.12

Does this code compile? If so what is its output?

```
import java.util.*;
public class Sq {
    public static void main(String[] args) {
        String[] s = new String[]{"a","b","c","d"};
        String b = new String("b");
        for(int i=0;i<s.length;i++)
            if(s[i] == b) System.out.print("-B-");
            else System.out.print(s[i]);
        System.out.println();
    }
}
```

abcd

1.13

Does this code compile? If so what is its output?

```
import java.util.*;
public class Sq {
    public static void main(String[] args) {
        String[] s = new String[]{"a","b","c","d"};
        String b = new String("b");
        for(int i=0;i<s.length;i++)
            if(s[i].equals(b)) System.out.print("-B-");
            else System.out.print(s[i]);
        System.out.println();
    }
}
```

```
}  
}
```

a-B-cd

1.14

Create a function named `lsq` that takes an argument `n`. The function should return an array of type `int` containing all the squares that are less than or equal to `n`. Thus, `lsq(25)` should return `[1,4,16,25]`.

```
public static int[] lsq(int n) {  
    int sz = 0;  
    for(int i=0;i*i <= n;i++) sz++;  
    int[] data = new int[sz];  
    for(int i=0;i*i <= n;i++)  
        data[i] = i*i;  
    return data;  
}
```

2 Classes and Inheritance (Ch. 8/9)

2.1

Does this code compile? If it does, what is its output?

```
class A {  
    public void foo() { System.out.println("Hello_A"); }  
}  
public class B extends A {  
    public void foo() { System.out.println("Hello_B"); }  
    public static void main(String[] args) {  
        A a = new B();  
        a.foo();  
    }  
}
```

Hello B

2.2

Does this code compile? If it does, what is its output?

```
class A {  
    public void foo() { System.out.println("Hello_A"); }  
}  
public class B implements A {  
    public void foo() { System.out.println("Hello_B"); }  
    public static void main(String[] args) {  
        A a = new B();  
        a.foo();  
    }  
}
```

compile error:

B.java:5: error: interface expected here

```
public class B implements A {  
    ^
```

B.java:8: error: incompatible types: B cannot be converted to A

```
    A a = new B();  
    ^
```

2 errors

2.3

Does this code compile? If it does, what is its output?

```
class A {
    public void foo() { System.out.println("Hello_A"); }
}
public class B extends A {
    private void foo() { System.out.println("Hello_B"); }
    public static void main(String[] args) {
        A a = new B();
        a.foo();
    }
}
```

compile error:

```
B.java:6: error: foo() in B cannot override foo() in A
    private void foo() { System.out.println("Hello_B"); }
           ^
    attempting to assign weaker access privileges; was public
1 error
```

2.4

Does this code compile? If it does, what is its output?

```
interface A {
    public void foo() { System.out.println("Hello_A"); }
}
public class B implements A {
    public void foo() { System.out.println("Hello_B"); }
    public static void main(String[] args) {
        A a = new B();
        a.foo();
    }
}
```

compile error:

```
B.java:3: error: interface abstract methods cannot have body
    public void foo() { System.out.println("Hello_A"); }
           ^
1 error
```

2.5

Does this code compile? If it does, what is its output?

```
interface A {
    public void foo();
}
public class B implements A {
    public void foo() { System.out.println("Hello_B"); }
    public static void main(String[] args) {
        A a = new B();
        a.foo();
    }
}
```

Hello B

2.6

Does this code compile? If it does, what is its output?

```
public class B {
    static int a;
    int b;
    public String toString() { return "("+a+", "+b+")"; }
    public static void main(String[] args) {
        B b = new B();
        b.a = 1; b.b = 2;
        B c = new B();
        c.a = 3; c.b = 4;
        System.out.println(b+" "+c);
    }
}
```

(3,2) (3,4)

2.7

Does this code compile? If it does, what is its output?

```
class A {
    static void foo() { System.out.println("Hello_A"); }
}
public class B extends A {
    static void foo() { System.out.println("Hello_B"); }
    public static void main(String[] args) {
        A a = new A();
        a.foo();
    }
}
```

Hello A

2.8

Does this code compile? If it does, what is its output?

```
class A { }
public class B extends A {
    public static void main(String[] args) {
        A a = new B();
        System.out.println("works");
    }
}
```

works

2.9

Does this code compile? If it does, what is its output?

```
class A { }
public class B extends A {
    public static void main(String[] args) {
        B a = new A();
        System.out.println("works");
    }
}
```

compile error:

```
B.java:5: error: incompatible types: A cannot be converted to B
    B a = new A();
           ^
1 error
```

2.10

Does this code compile? If it does, what is its output?

```
class A { }
public class B extends A {
    public static void main(String[] args) {
        B a = (B)new A();
        System.out.println("works");
    }
}
```

ClassCastException

2.11

Does this code compile? If it does, what is its output?

```
class A { }
public class B extends A {
    public static void main(String[] args) {
        A a = new B();
        B b = (B)a;
        System.out.println("works");
    }
}
```

works

2.12

Does this code compile? If it does, what is its output?

```
class A {
    int val1() { return 3; }
    int val2() { return 4; }
}
public class B extends A {
    int val1() { return 5; }
    public static void main(String[] args) {
        A a1 = new B();
        A a2 = new A();
        System.out.println("Num="+a1.val1()*a2.val2());
    }
}
```

Num=20

2.13

Does this code compile? If it does, what is its output?

```
class A {
    boolean isSuper() { return true; }
    void show() {
        if(isSuper()) System.out.println("Super!");
        else System.out.println("Not super.");
    }
}
```



```

public class B extends A {
    boolean isSuper() { return false; }
    public static void main(String[] args) {
        A a = new A();
        a.show();
        a = new B();
        a.show();
    } }

```

Super!
Not super.

2.14

Does this code compile? If it does, what is its output?

```

class A {
    boolean isSuper() { return true; }
}
public class B extends A {
    boolean isSuper() { return false; }
    public static void main(String[] args) {
        A a1 = new A();
        A a2 = new B();
        B b = null;
        if(!a1.isSuper()) b = (B)a1;
        if(!a2.isSuper()) b = (B)a2;
        System.out.println("Found␣B");
    } }

```

Found B

2.15

Does this code compile? If it does, what is its output?

```

class A {
    boolean isSuper() { return true; }
}
public class B extends A {
    boolean isSuper() { return false; }
    public static void main(String[] args) {
        A a1 = new A();
        A a2 = new B();
        B b = null;
        if(a1.isSuper()) b = (B)a1;
        if(a2.isSuper()) b = (B)a2;
        System.out.println("Found␣B");
    } }

```

ClassCastException

2.16

Does this code compile? If it does, what is its output?

```

class A { }
public class B extends A {
    public static void main(String[] args) {
        A a1 = new A();
        A a2 = new B();
        B b = null;
        if(a1 instanceof B) b = (B)a1;
    } }

```

```

    if(a2 instanceof B) b = (B)a2;
    System.out.println("Found B");
} }

```

Found B

3 Anonymous Inner Classes (Covered with Ch. 8/9)

The java.lang package has a class called Runnable.

```

package java.lang;
public interface Runnable {
    public void run();
}

```

3.1

Write a complete “Hello, World” program in Java that instantiates an anonymous inner class based on the Runnable interface, then calls the instantiated object to print the message.

```

public class HelloRun {
    public static void main(String[] args) {
        Runnable r = new Runnable() {
            public void run() {
                System.out.println("Hello, World");
            }
        };
        r.run();
    }
}

```

3.2

Rewrite the following code to make use of an anonymous inner class

```

class MyAdapater extends MouseAdapter {
    public void mousePressed(MouseEvent me) {
        System.out.println("Press"); } }
...
public class MyApp {
    public static void main(String[] args) {
        JFrame bf = new JFrame("MyApp");
        bf.addMouseListener(new MyAdapter());
    }
}

```

```

public class MyApp {
    public static void main(String[] args) {
        JFrame bf = new JFrame("MyApp");
        bf.addMouseListener(new MouseAdapter() {
            public void mousePressed(MouseEvent me) {
                System.out.println("Press"); }
        });
    }
}

```