Which of the following statements will not compile because of an incorrect type cast?
Select the one correct answer.
A. int i0 = (char) '$';
B. int i1 = (int) 76.1234;
C. int i2 = (short) 34;
D. int i3 = (float) 3213;
E. All of the above

What is the value of myInteger after execution of the following code?
```
int myInteger = 1;
if (--myInteger == 1) {
    myInteger += 4;
} else {
    myInteger += 5;
}
```
A. 3
B. 4
C. 5
D. 6
E. None of the above

What does the code shown below print to the console?
```
System.out.print(Math.toDegrees(Math.PI * 2.0) + ",");
System.out.print(Math.sqrt(64.0) + ",");
System.out.printf("%.4f\n", Math.min(Math.PI, 1.2345));
```
A. 180.0, 64.0, 3.1416
B. 180.0, 64.0, 1.2345
C. 360.0, 8.0, 3.1416
D. 360.0, 8.0, 1.2345

What does the following code print for the final value of the loop variable?
```java
char c;
for (c = 'A'; c <= 'S'; c++) {
    System.out.println("c is " + c);
}
```
A. c is R
B. c is S
C. c is T
D. Will not compile!

What are the limitations of single return value from a method?
A. A single primitive (byte, int, float, double, char, boolean, ...)
B. A single primitive or an array of primitives
C. A single class, (String, Scanner, ...)
D. A single class or an array of classes
E. All of the above
Which are the correct values of `iArray` after running the code fragment below?

```java
// Code fragment
int iArray[] = {2, 3, 4, 5, 6};
myMethod(iArray);
// Method definition
static void myMethod(int array[]) {
    for (int i=1; i<array.length-1; i++)
        array[i] *= 2;
}
```

A. 2, 3, 4, 5, 6  
B. 2, 6, 8, 10, 6  
C. 2, 6, 8, 10, 12  
D. 4, 6, 8, 10, 6  
E. None of the above

---

Ragged 2D Arrays

What are the values of `iArray.length` and `iArray[1].length`, and `iArray[3].length`?

```java
int iArray[][] = {
    {11, 22, 33, 44},
    {55, 66, 77},
    {88, 99},
    {1, 2, 3, 4, 5, 6}
};
```

A. 4, 3, 6  
B. 4, 4, 5  
C. 14, 3, 6  
D. 14, 7, 14  
E. 3, 2, 5

---

Which of the following statements about static and non-static are correct?

A. Static data is also called instance data, and non-static data is called class data.  
B. Only one copy of instance (non-static) exists.  
C. There is a separate copy of instance data for every object that is instantiated.  
D. Accessing class data using the class name instead of the object name is not a good practice.  
E. Accessing instance data does not require use of the class name, if done from within the same class.

---

Given the file contents in red and a Scanner `in`, which values are read to `d0/d1/d2`?

```
11.1 83 22.2 33.3 1234
```

double do = in.nextDouble();
double d1 = in.nextDouble();
double d2 = in.nextDouble();

A. 11.1, 22.2, 33.3  
B. 11, 83, 22  
C. 11.1, 83.0, 22.2  
D. 11.1, exception!  

---

Select the correct definition of the usage of a Java abstract class.

A. An abstract class provides shared code and data for a set of classes that share attributes and behaviors.  
B. An abstract class is similar to an interface in that it specifies functionality, but has no actual code or data.  
C. An abstract class differs from an interface in that it must implement every method that it contains.  
D. An abstract class can be instantiated, but code for its abstract methods might be missing.

---

What does the recursive code below print when called with `s = "aabbccddeeff"`?

```java
public static String munge(String s) {
    if (s == null || s.length() <= 1) // base case
        return s;
    else if (s.charAt(0) == s.charAt(1))
        return munge(s.substring(1, s.length()));
    else
        return s.charAt(0) +
               munge(s.substring(1, s.length()));
}
```

A. aabbcdddef  
B. abcdedfabc  
C. aabcdcdefab  
D. abcdedefab  
E. empty string

---

Recursion Example
What is the contents of the ArrayList after the following code executes?

```java
ArrayList<String> list = new ArrayList<>();
list.add("Java");
list.add("Basic");
list.add("C++");
list.add(2, "Python");
list.add(3, "Fortran");
list.remove(1);
```

A. [Java, Basic, Python, Fortran, C++]
B. [Java, Python, C++, Fortran]
C. [Basic, Python, Fortran, C++]
D. [Java, Python, Fortran, C++]