

Elements of Java, 1

Assume some code is included that declares `a` as an array of 6 ints, and consider the following code segment:

```
int i;  
int sum = 0;  
int positives = 0;  
  
a[0] = 2;  
a[1] = -7;  
a[2] = 1;  
a[3] = -8;  
a[4] = -2;  
a[5] = 8;  
  
for (i = 0; i < 6; i++)  
{ sum += a[i];  
  if (a[i] > 0)  
    positives++;  
}
```

Does this code

- A. Compile and run in both C and Java
- B. Compile and run in C, not Java
- C. Compile and run in Java, not C
- D. Fail to compile and run in both C and Java

Elements of Java, 2

We create a project directory Project1.

Within Project1, we create a package directory pack1

Within pack1, we create a Java class program, prog1.java, that explicitly identifies that it is within pack 1

To compile and run prog1, three approaches are proposed for a terminal window:

1. In the Project1 directory, we type

```
javac prog1.java  
java prog1
```

2. In the Project1 directory, we type

```
javac pack1/prog1.java  
java pack1/prog1
```

3. In the pack1 directory, we type

```
javac prog1.java  
java prog1
```

Which of these approaches works:

A. None

B. only 1 (not 2 or 3)

C. only 2 (not 1 or 3)

D. only 3 (not 1 or 2)

E. some combination of 2 or more

Elements of Java, 3

Consider the following Java program:

```
public class Intro2
{
    public static void main ()
    {
        System.out.println ("hello, ");
        System.out.println ("CSC 161");
    }
}
```

What happens if we try to compile and run this program

- A. A compilation error is reported
- B. Program compiles, but a runtime error is reported
- C. Program compiles, and the program prints hello,CSC 161
- D. Program compiles, and the program prints hello, CSC 161
- E. Program compiles, and the program prints hello,
CSC 161

Elements of Java, 4

Consider the following program

```
public class Intro3
{
    public static void main ()
    { System.out.println ("hello, ");
      System.out.println ("CSC 161");
    }
}
```

```
public static void main (String arg [ ])
{ System.out.println ("Here we go");
  if (arg.length == 0)
  { System.out.println ("no strings supplied");
    String [ ] myStrings = {"a", "b123", "c"};
    main (myStrings);
  }
  else
  { System.out.println ("the number of strings supplied is "+arg.length);
    System.out.println (" these strings are");
    for (int i = 0; i < arg.length; i++)
      System.out.println ("    " + arg[i]);
    main ();
  }
}
```

What happens if we try to compile/run:

- A. Does not compile
- B. Compiles, but does not run
- C. Compiles/runs, output may vary
- D. some, not all println statements utilized
- E. all println statements utilized

Elements of Java, 5

In a Java program, we want to declare and use an array a with 6 int values.

Which of the following could be used to declare a:

- 1. `int [6] a;`
- 2. `int [ ] a;`

- | | |
|--------------------|-----------------|
| A. Neither 1 nor 2 | C. 2, but not 1 |
| B. 1, but not 2 | D. Both 1 and 2 |

Once declared, we must be sure a is initialized; two approaches are proposed

- 3. No further initialization needed
- 4. `a = new int [6];`

- | | |
|--------------------|-----------------|
| A. Neither 3 nor 4 | C. 4, but not 3 |
| B. 3, but not 4 | D. Both 3 and 4 |

With this initialization, what happens if the following line is encountered

`System.out.println ("a[4] = " + a[4]);`

- | | |
|--|---|
| A. A compiler error results | C. The program compiles and prints 0 |
| B. Execution halts with some type of error | D. The program compiles and prints an unknown value |

Same question for

`System.out.println ("a[7] = " + a[7]);`