AP Computer Science
Chapter 8 Test Review

1   Multiple-Choice Questions

For the questions below, assume values is an int array that is currently filled to capacity, with the following values:

9	4	12	2	6	8	18

1) What is returned by values[3]?
A) 9
B) 12
C) 2
D) 6
E) 3

Answer: C

2) What is the value of values.length?
A) 0
B) 5
C) 6
D) 7
E) 18

Answer: D

3) Which of the following loops would adequately add 1 to each element stored in values?
A) `for (j=1; j<values.length; j++)  values[j]++;`
B) `for (j=0; j<values.length; j++)  values[j]++;`
C) `for (j=0; j<=values.length; j++)  values[j]++;`
D) `for (j=0; j<values.length-1; j++) values[j]++;`
E) `for (j=1; j<values.length-1; j++)  values[j]++;`

Answer: B

4) The statement `System.out.println(values[7]);` will
A) output 7
B) output 18
C) output nothing
D) cause an ArrayOutOfBoundsException to be thrown
E) cause a syntax error

Answer: D

5) Which of the following is a legal way to declare and instantiate an array of 10 Strings?
A) `String s = new String(10);`
B) `String[10] s = new String;`
C) `String[ ] s = new String[10];`
D) `String s = new String[10];`
E) `String[ ] s = new String;`

Answer: C

6) In Java, arrays are
A) primitive data types
B) objects
C) interfaces
D) primitive data types if the type stored in the array is a primitive data type and objects if the type stored in the array is an object
E) Strings

Answer: B

7) The "off-by-one" error associated with arrays arises because
A) the first array index is 0 and programmers may start at index 1, or may use a loop that goes one index too far
B) the last array index is at length + 1 and loops may only iterate to length, missing one
C) the last array element ends at length - 1 and loops may go one too far
D) programmers write a loop that goes from 0 to length - 1 whereas the array actually goes from 1 to length
E) none of the above, the "off-by-one" error has nothing to do with arrays

Answer: A

8) What does the following code do? Assume list is an array of `int` values, temp is some previously initialized `int` value, and c is an `int` initialized to 0.

```
      for (int j = 0; j < list.length; j++)
            if (list[j] < temp) c++;
```

A) It finds the smallest value and stores it in temp
B) It finds the largest value and stores it in temp
C) It counts the number of elements equal to the smallest value in list
D) It counts the number of elements in list that are less than temp
E) It sorts the values in list to be in ascending order

Answer: D

For the questions #9 ~ 11 below, assume an int array, candy, stores the number of candy bars sold by a group of children where candy[j] is the number of candy bars sold by child j. Assume there are 12 children in all.

9) What does the following code do?

```
            Scanner scan = Scanner.create(System.in);
            int value1 = scan.nextInt( );
            int value2 = scan.nextInt( );
            bars[value1] += value2;
```

A) adds 1 to the number of bars sold by child value1 and child value2
B) adds 1 to the number of bars sold by child value1
C) adds value1 to the number of bars sold by child value2
D) adds value2 to the number of bars sold by child value1
E) inputs a new value for the number of bars sold by both child value1 and child value2

Answer: D

10) Which of the following code could be used to compute the total number of bars sold by the children?
A) `for (int j=0; j<12; j++) sum += candy[j];`
B) `for (int j=0; j<12; j++) candy[j] = sum;`
C) `for (int j=0; j<12; j++) sum = candy[j];`
D) `for (int j=0; j<12; j++) sum += [j];`
E) `for (int j=0; j<12; j++) [j] += sum;`

Answer: A

11) What does the following method do?

```
        public int question15( )
        {
              int value1 = 0;
              int value2 = 0;
              for (int j = 0; j < 12; j++)
                 if (candy[j] > value1)
                 {
                       value1 = candy[j];
                       value2 = j;
                 }
              return value2;
        }
```

A) It returns the total number of candy bars sold
B) It returns the total number of children who sold 0 candy bars
C) It returns the total number of children who sold more than 0 candy bars
D) It returns the number of candy bars sold by the child who sold the most candy bars
E) It returns the index of the child who sold the most candy bars

Answer: E

12) Consider the array declaration and instantiation: `int[ ] arr = new int[5];` Which of the following is true about `arr`?
A) It stores 5 elements with legal indices between 1 and 5
B) It stores 5 elements with legal indices between 0 and 4
C) It stores 4 elements with legal indices between 1 and 4
D) It stores 6 elements with legal indices between 0 and 5
E) It stores 5 elements with legal indices between 0 and 5

Answer: B

13) If an int array is passed as a parameter to a method, which of the following would adequately define the parameter list for the method header?
A) `(int[ ])`
B) `(int a[ ])`
C) `(int[ ] a)`
D) `(int a)`
E) `(a[ ])`

Answer: C

14) If `int[ ] x = new int[15];` and the statement `x[-1] = 0;` is executed, then which of the following Exceptions is thrown?
A) `IndexOutOfBoundsException`
B) `ArrayIndexOutOfBoundsException`
C) `NegativeArraySizeException`
D) `NullPointException`
E) `ArithmeticException`

Answer: B

15) Assume that `BankAccount` is a predefined class and that the declaration `BankAccount[ ] firstEmpireBank`; has already been performed. Then the following instruction reserves memory space for

```
firstEmpireBank = new BankAccount[1000];
```

A) a reference variable to the memory that stores all 1000 `BankAccount` entries
B) 1000 reference variables, each of which point to a single `BankAccount` entry
C) a single `BankAccount` entry
D) 1000 `BankAccount` entries
E) 1000 reference variables and 1000 `BankAccount` entries

Answer: B

16) The following code accomplishes which of the tasks written below? Assume list is an `int` array that stores positive `int` values only.

```
int foo = 0;
for (int j =0 ; j < list.length; j++)
  if (list[j] > foo) foo = list[j];
```

A) It stores the smallest value in list (the minimum) in foo
B) It stores the largest value in list (the maximum) in foo
C) It stores every value in list, one at a time, in foo, until the loop terminates
D) It counts the number of elements in list that are greater than foo
E) It counts the number of elements in list that are less than foo

Answer: B

17) If `x` is a `char`, and values is an `int` array, then `values[x]`
A) causes a syntax error
B) causes an Exception to be thrown
C) casts x as an `int` based on x's position in the alphabet (for instance, if x is 'a' then it uses 0 and if x is 'z' then it uses 25)
D) casts x as an `int` based on x's ASCII value (for instance, if x is 'a' then it uses 97 and if x is 'z' then it uses 122)
E) casts x as an `int` based on the digit that is stored in x (for instance, if x is '3' it uses 3) but throws an exception if x does not store a digit

Answer: D

18) Given the following declarations, which of the following variables are arrays?

```
int[ ] a, b;
int c, d[ ];
```

A) a
B) a and b
C) a and d
D) a, b and d
E) a, b, c and d

Answer: D

2 True/False Questions

1) Arrays have a built in toString method that returns all of the elements in the array as one String with `"\n"` inserted between each element.

Answer: FALSE

2) Java arrays can store primitive types and Strings, but cannot store any other type of Object other than Strings.

Answer: FALSE

3) A Java main method uses the parameter (`String[ ] variable`) so that a user can run the program and supply "command-line" parameters. Since the parameter is a String array, however, the user does not have to supply any parameters.

Answer: TRUE

4) An array index cannot be a `float`, `double`, `boolean` or `String`.

Answer: TRUE

5) If the following statement is performed: `CD[ ] mycollection = new CD[200];` where `CD` is a previously defined class, then `mycollection[5]` is a `CD` object.

Answer: TRUE

6) It is possible to sort an array of `int`, `float`, `double` or `String`, but not an array of an Object class such as a `CD` class.

Answer: FALSE

7) To swap the 3rd and 4th elements in the `int` array values, you would do:

```
        values[3] = values[4];
        values[4] = values[3];
```

Answer: FALSE

8) In Java, an array can only store one type of data. For instance, you cannot create an array that stores both double and String values.

Answer: TRUE

9) In a two-dimensional array, both dimensions must have the same number of elements, as `in[10][10]`.

Answer: FALSE

10) An array, when instantiated, is fixed in size, but an `ArrayList` can dynamically change in size when new elements are added to it.

Answer: TRUE

3 Free-Form/Short Answer Questions

1) Write a method to compute the average of an `int` array and return the value as a `double`. The `int` array and the number of elements in the array are both passed as parameters to the method in that order.

Answer:
```
public double computeAverage(int[ ] list, int n)
{
    int sum = 0;
    for (int j = 0; j < n; j++)
        sum += list[j];
    return (double) ((n > 0) ? sum / n : 0);
}
```

2) The length operator can be used to control a `for`-loop that iterates through each element of an array, as in

```
              for (int j=0; j<list.length; j++)
```

However, this is not necessarily safe. Why not?

Answer: It is not necessarily the case that every element in the array has a value stored there. So, list[j] might be an array element that has no value even though j < list.length. So, list.length can only be used safely in an array that is full.

3) Write a method to compute and return the value of max - min where max is the largest element of an `int` array and min is the smallest element of an `int` array. The method is passed the `int` array and an `int` value that denotes the number of elements in the array. You may assume that the `int` array has at least 1 element in it.

Answer:
```
public int computeDifference(int[ ] values, int n)
{
      int min = values[0];
      int max = values[0];
      for (int j = 1; j < n; j++)
      {
           if (values[j] < min) min = values[j];
           if (values[j] > max) max = values[j];
      }
      return max - min;
}
```

4) Write code fragment to swap the two Strings stored by variables `a` and `b`.

Answer:
```
String temp = a;
a = b;
b = temp;
```

5) What is the output of the following code snippet?

```
public static void main(String[] args)
```

```
{
   String[] arr = { "aaa", "bbb", "ccc" };
   mystery(arr);
   System.out.println(arr[0] + " " + arr.length);
}
public static void mystery(String[] arr)
{
   arr = new String[5];
   arr[0] = "ddd";
}
```

Answer: `aaa 3`

6) Write a code fragment to create a two-dimensional 10x10 array and initialize every element to be the value of `i * j` where `i` and `j` are the `two` indices (for instance, `element [5][3] is 5 * 3 = 15`).

Answer:

```
int[ ][ ] matrix = new int[10][10];
for (int j=0; j<10; j++)
   for (int k=0; k<10; k++)
      matrix[j][k] = j*k;
```

7) Write a method that takes an array of `Strings` as a parameter and the number of elements currently stored in the array, and returns a single `String` that is the concatenation of all `Strings` in the array.

Answer:

```
public String arrayConcatenator(String[ ] list, int number)
{
     String allElements = "";
     for (int j=0; j<number; j++)
          allElements += list[j];
     return allElements;
}
```