

## Unit 6 Practice Test

1. Given the array of integers `arr` shown below

13	7	27	2	18	33	9	11	22	8
----	---	----	---	----	----	---	----	----	---

what is the output of the following statements?

```
int[] p = new int[10];
int[] q = new int[10];
for (int i = 0; i < 10; i++)
    p[i] = arr[i];
q = p;
p[4] = 20;
System.out.println(arr[4] + " " + q[4]);
```

- A. 20 20
  - B. 18 18
  - C. 2 2
  - D. 18 20
  - E. 2 20
2. Consider the following method:

```
public static int test(int a, int b) {
    if (a < b)
        return 0;
    else
        return (1 + test(a-b, b));
}
```

What is returned by the call `test(15, 4)`?

- A. 1
  - B. 2
  - C. 3
  - D. 4
  - E. 5
3. Which of the following statements about memory management in Java is *not* true?
- A. Local variables and method parameters are stored on the stack.
  - B. Memory on the heap is allocated dynamically at runtime.
  - C. Objects allocated on the heap persist until there are no remaining references to them.
  - D. Static memory is used for the instance variables of a class.
  - E. The contents of an array can never be stored on the stack.

4. A program for recursive backtracking includes a method similar to this one:

```
public void key_function(int i) {
    for (int alternative = i; alternative < n;
        alternative++)
    {
        do_alt(alternative);
        *****
        undo_alt(alternative);
    }
}
```

The line \*\*\*\*\* should be replaced by:

- A. do\_alt(alternative + 1);
  - B. key\_function(alternative);
  - C. key\_function(alternative + 1);
  - D. key\_function(i);
  - E. key\_function(i + 1);
5. Assume that the following method has been added to the ArrayBag class:

```
public static void changeOneItem(Object newItem) {
    items[0] = newItem;
}
```

Which of the following statements are true?

- i. Because the method is static, we would need to use the class name to invoke it from outside the ArrayBag class (e.g., ArrayBag.changeFirst(...)).
  - ii. The method will generate an error at compile time.
  - iii. When the method is invoked, the parameter newItem will contain a copy of the object that is passed in as an argument.
- A. only i is true
  - B. only ii is true
  - C. only iii is true
  - D. i and ii are true, but iii is not
  - E. i, ii, and iii are all true

6. Write a *recursive* method named `sumReciprocals` that takes as its only argument a non-negative integer, `n`, and returns a double value that is the sum of the reciprocals of the integers from 1 to `n`. For example, `sumReciprocals(2)` should return 1.5, which is  $1/1 + 1/2$ , and `sumReciprocals(4)` should return approximately 2.0833, which is  $1/1 + 1/2 + 1/3 + 1/4$ . You do *not* need to perform any error checking on the value of the parameter. No use of iteration is allowed.

```
public static double sumReciprocals(int n)
```

7. Write a method for the `ArrayBag` class with the following signature:

```
public int count(Object item)
```

It should return the number of times that the specified `item` occurs in the `ArrayBag` on which the method is invoked. For example, if `b` is an `ArrayBag` that represents the bag `{5, 7, 2, 10, 7}`, `b.count(7)` should return 2, `b.count(10)` should return 1, and `b.count(20)` should return 0. Your method does *not* need to use recursion.