

1. Consider this inheritance hierarchy, in which `Novel` and `Textbook` are subclasses of `Book`.

Basic
1

- Which of the following is true?
- (A) The `Textbook` class has methods that are in neither `Book` nor `Novel`.
 - (B) Each of `Novel` and `Textbook` have a method `computeSales` that is identical, but different from `Book`.
 - (C) If the `Book` class has a method `getAuthor`, then `Novel` and `Textbook` also have a method `getAuthor`.
 - (D) Both `Novel` and `Textbook` inherit the `readFile` method from `Book`.
 - (E) If the `Book` class has a private method called `readFile`, this method may not be accessed in either the `Novel` or `Textbook` classes.

2. A programmer is designing a program to catalog all books in a library. He plans to have a `Book` class that stores features of each book: `author`, `title`, `isOnShelf`, and so on, with operations like `getAuthor`, `getTitle`, `getShelfInfo`, and `setShelfInfo`. Another class, `LibraryList`, will store an array of `Book` objects. The `LibraryList` class will include operations such as `listAllBooks`, `addBook`, `removeBook`, and `searchForBook`. The programmer plans to implement and test the `Book` class first, before implementing the `LibraryList` class. The programmer's plan to write the `Book` class first is an example of
- (A) top-down development.
 - (B) bottom-up development.
 - (C) procedural abstraction.
 - (D) information hiding.
 - (E) a driver program.

3. The color of a pixel can be represented using the RGB (Red, Green, Blue) color model, which stores values for red, green, and blue, each ranging from 0 to 255. How many bits (binary digits) would be needed to represent a color in the RGB model?
- (A) 8
 - (B) 16
 - (C) 24
 - (D) 32
 - (E) 40

10. The boolean expression `a[i] == max || !(max != a[i])` can be simplified to
- (A) `a[i] == max`
 - (B) `a[i] != max`
 - (C) `a[i] < max || a[i] > max`
 - (D) `true`
 - (E) `false`

11. Suppose the characters 0, 1, ..., 8, 9, A, B, C, D, E, F are used to represent a hexadecimal (base-16) number. Here A = 10, B = 11, ..., F = 15. What is the largest base-10 integer that can be represented with a two-digit hexadecimal number, such as 14 or 3A?
- (A) 32
 - (B) 225
 - (C) 255
 - (D) 256
 - (E) 272

20. Consider the following method.

```
public static void whatsIt(int n)
{
    if (n > 10)
        whatsIt(n / 10);
    System.out.print(n % 10);
}
```

What will be output as a result of the method call `whatsIt(347)`?

- (A) 74
- (B) 47
- (C) 734
- (D) 743
- (E) 347

23. Consider the code segment

```
if (n == 1)
    k++;
else if (n == 4)
    k += 4;
```

Suppose that the given segment is rewritten in the form

```
if (/* condition */)
    /* assignment statement */;
```

Given that n and k are integers and that the rewritten code performs the same task as the original code, which of the following could be used as

- (1) */* condition */* and (2) */* assignment statement */*?
- (A) (1) $n == 1 \ \&\& \ n == 4$ (2) $k += n$
- (B) (1) $n == 1 \ \&\& \ n == 4$ (2) $k += 4$
- (C) (1) $n == 1 \ || \ n == 4$ (2) $k += 4$
- (D) (1) $n == 1 \ || \ n == 4$ (2) $k += n$
- (E) (1) $n == 1 \ || \ n == 4$ (2) $k = n - k$

24. Which of the following will execute *without* throwing an exception?

I String s = null;
String t = "";
if (s.equals(t))
System.out.println("empty strings?");

II String s = "holy";
String t = "moly";
if (s.equals(t))
System.out.println("holy moly!");

III String s = "holy";
String t = s.substring(4);
System.out.println(s + t);

- (A) I only
(B) II only
(C) III only
(D) I and II only
(E) II and III only

30. Refer to method insert described here. The insert method has two string parameters and one integer parameter. The method returns the string obtained by inserting the second string into the first starting at the position indicated by the integer parameter pos. For example, if str1 contains xy and str2 contains cat, then

```
insert(str1, str2, 0) returns catxy
insert(str1, str2, 1) returns xcaty
insert(str1, str2, 2) returns xycat
```

Method insert follows:

```
/** Precondition: 0 <= pos <= str1.length().
 * Postcondition: If str1 = a0a1...an-1 and str2 = b0b1...bm-1,
                  returns a0a1...apos-1b0b1...bm-1aposapos+1...an-1
public static String insert(String str1, String str2, int pos)
{
    String first, last;
    /* more code */
    return first + str2 + last;
}
```

Which of the following is a correct replacement for `/* more code */`?

- (A) `first = str1.substring(0, pos);`
`last = str1.substring(pos);`
- (B) `first = str1.substring(0, pos - 1);`
`last = str1.substring(pos);`
- (C) `first = str1.substring(0, pos + 1);`
`last = str1.substring(pos + 1);`
- (D) `first = str1.substring(0, pos);`
`last = str1.substring(pos + 1, str1.length());`
- (E) `first = str1.substring(0, pos);`
`last = str1.substring(pos, str1.length() + 1);`

1. For ticket-selling purposes, there are three categories at a certain theater:

<u>Age</u>	<u>Category</u>
65 or above	Senior
From 18 to 64 inclusive	Adult
Below 18	Child

Which of the following code segments will assign the *correct* string to `category` for a given integer `age`?

```
I if (age >= 65)
    category = "Senior";
if (age >= 18)
    category = "Adult";
else
    category = "Child";
```

```
II if (age >= 65)
    category = "Senior";
if (18 <= age <= 64)
    category = "Adult";
else
    category = "Child";
```

```
III if (age >= 65)
    category = "Senior";
else if (age >= 18)
    category = "Adult";
else
    category = "Child";
```

- (A) I only
- (B) II only
- (C) III only
- (D) II and III only
- (E) I, II, and III

2. What is the output of the following code segment?

```
String s = "How do you do?";
int index = s.indexOf("o");
while (index >= 0)
{
    System.out.print(index + " ");
    s = s.substring(index + 1);
    index = s.indexOf("o");
}
```

- (A) 1 3 2 3
- (B) 2 4 3 4
- (C) 1 5 8 12
- (D) 1 5 8 11
- (E) No output because of an `IndexOutOfBoundsException`

3. Consider the following method `removeAll` that creates and returns a string that has stripped its input phrase of all occurrences of its single-character `String` parameter `ch`.

```
Line 1: public static String removeAll(String phrase, String ch)
Line 2: {
Line 3:     String str = "";
Line 4:     String newPhrase = phrase;
Line 5:     int pos = phrase.indexOf(ch);
Line 6:     if (pos == -1)
Line 7:         return phrase;
Line 8:     else
Line 9:     {
Line 10:         while (pos >= 0)
Line 11:         {
Line 12:             str = str + newPhrase.substring(0, pos - 1);
Line 13:             newPhrase = newPhrase.substring(pos + 1);
Line 14:             pos = newPhrase.indexOf(ch);
Line 15:             if (pos == -1)
Line 16:                 str = str + newPhrase;
Line 17:         }
Line 18:         return str;
Line 19:     }
Line 20: }
```

The method doesn't work as intended. Which of the following changes to the `removeAll` method will make it work as specified?

- (A) Change Line 10 to
`while (pos >= -1)`
- (B) Change Line 12 to
~~`str = str + newPhrase.substring(0, pos);`~~
- (C) Change Line 13 to
`newPhrase = newPhrase.substring(pos);`
- (D) Change Line 14 to
`pos = phrase.indexOf(ch);`
- (E) Change Line 16 to
`str = str + newPhrase.substring(pos + 1);`

4. A programmer has written a program that "chats" to a human user based on statements that the human inputs. The program contains a method `findKeyword` that searches an input statement for a given keyword. The `findKeyword` method contains the following line of code:

```
pos = statement.indexOf(word);
```

Suppose `pos` has a value ≥ 0 , that is, `word` was found. The programmer now wants to test that an actual word was found, not part of another word. For example, if "cat" is the keyword, the programmer needs to check that it's not part of "catch" or "category." Here is the code that tests if `word` is a stand-alone word. (You may assume that `statement` is all lowercase and contains only letters and blanks.)

```
pos = statement.indexOf(word);
//Check for first or last word
if (pos == 0 || pos + word.length() == statement.length())
{
    before = " ";
    after = " ";
}
else
{
    before = statement.substring(pos - 1, pos);
    after = statement.substring(pos + word.length(),
        pos + word.length() + 1);
    if (/* test */)
        //then a stand-alone word was found ...
    else
        //word was part of a larger word
}
```

Which replacement for `/* test */` will give the desired result?

- (A) `(before < "a" || before > "z") && (after < "a" || after > "z")`
- (B) `(before > "a" || before < "z") && (after > "a" || after < "z")`
- (C) `(before.compareTo("a") < 0 && before.compareTo("z") > 0) || (after.compareTo("a") > 0 && after.compareTo("z") < 0)`
- (D) `(before.compareTo("a") > 0 && before.compareTo("z") < 0) && (after.compareTo("a") > 0 && after.compareTo("z") < 0)`
- (E) `(before.compareTo("a") < 0 || before.compareTo("z") > 0) && (after.compareTo("a") < 0 || after.compareTo("z") > 0)`

5. A program that simulates a conversation between a computer and a human user generates a random response to a user's comment. All possible responses that the computer can generate are stored in an array of String called allResponses. The method given below, getResponse, returns a random response string from the array.

```
/** Precondition: array allResponses is initialized with strings.  
 * Postcondition: returns a random response from allResponses.  
 */  
public String getResponse();  
{ /* implementation */ }
```

Which is a correct */* implementation */*?

- (A) int i = (int) (Math.random() * allResponses.length);
return allResponses[i];
- (B) return (String) (Math.random() * allResponses.length);
- (C) int i = Math.random() * allResponses.length;
return allResponses[i];
- (D) int i = (int) (Math.random() * (allResponses.length - 1));
return allResponses[i];
- (E) return (int) (Math.random() * allResponses.length);

1. A large Java program was tested extensively, and no errors were found. What can be concluded?
- (A) All of the preconditions in the program are correct.
 - (B) All of the postconditions in the program are correct.
 - (C) The program may have bugs.
 - (D) The program has no bugs.
 - (E) Every method in the program may safely be used in other programs.

17. An algorithm for finding the average of N numbers is

$$\text{average} = \frac{\text{sum}}{N}$$

where N and sum are both integers. In a program using this algorithm, a programmer forgot to include a test that would check for N equal to zero. If N is zero, when will the error be detected?

- (A) At compile time
 - (B) At edit time
 - (C) As soon as the value of N is entered
 - (D) During run time
 - (E) When an incorrect result is output
22. If a , b , and c are integers, which of the following conditions is sufficient to *guarantee* that the expression
- $$a < c \ || \ a < b \ \&\& \ !(a == c)$$
- evaluates to true?
- (A) $a < c$
 - (B) $a < b$
 - (C) $a > b$
 - (D) $a == b$
 - (E) $a == c$
23. Airmail Express charges for shipping small packages by integer values of weight. The charges for a weight w in pounds are as follows:

$0 < w \leq 2$	\$4.00
$2 < w \leq 5$	\$8.00
$5 < w \leq 20$	\$15.00

The company does not accept packages that weigh more than 20 pounds. Which of the following represents the best set of data (weights) to test a program that calculates shipping charges?

- (A) 0, 2, 5, 20
- (B) 1, 4, 16
- (C) -1, 1, 2, 3, 5, 16, 20
- (D) -1, 0, 1, 2, 3, 5, 16, 20, 22
- (E) All integers from -1 through 22

27. This question is based on the following declarations:

```
String strA = "CARROT", strB = "Carrot", strC = "car";
```

Given that all uppercase letters precede all lowercase letters when considering alphabetical order, which is true?

- (A) `strA.compareTo(strB) < 0 && strB.compareTo(strC) > 0`
- (B) `strC.compareTo(strB) < 0 && strB.compareTo(strA) < 0`
- (C) `strB.compareTo(strC) < 0 && strB.compareTo(strA) > 0`
- (D) `!(strA.compareTo(strB) == 0) && strB.compareTo(strA) < 0`
- (E) `!(strA.compareTo(strB) == 0) && strC.compareTo(strB) < 0`

34. Refer to the `nextIntInRange` method below:

```
/** @return a random integer in the range low to high, inclusive */
public int nextIntInRange(int low, int high)
{
    return /* expression */
}
```

Which `/* expression */` will always return a value that satisfies the postcondition?

- (A) `(int) (Math.random() * high) + low;`
- (B) `(int) (Math.random() * (high - low)) + low;`
- (C) `(int) (Math.random() * (high - low + 1)) + low;`
- (D) `(int) (Math.random() * (high + low)) + low;`
- (E) `(int) (Math.random() * (high + low - 1)) + low;`

28. A programmer has a file of names. She is designing a program that sends junk mail letters to everyone on the list. To make the letters sound personal and friendly, she will extract each person's first name from the name string. She plans to create a parallel file of first names only. For example,

fullName	firstName
Ms. Anjali DeSouza	Anjali
Dr. John Roufaiel	John
Mrs. Mathilda Concia	Mathilda

Here is a method intended to extract the first name from a full name string.

```
/** Precondition:
 * - fullName starts with a title followed by a period.
 * - A single space separates the title, first name, and last name.
 * @param fullName a string containing a title, period, blank,
 * and last name
 * @return the first name only in fullName
 */
public static String getFirstName(String fullName)
{
    final String BLANK = " ";
    String temp, firstName;

    /* code to extract first name */

    return firstName;
}
```

Which represents correct `/* code to extract first name */`?

- I int k = fullName.indexOf(BLANK);
temp = fullName.substring(k + 1);
k = temp.indexOf(BLANK);
firstName = temp.substring(0, k);
- II int k = fullName.indexOf(BLANK);
firstName = fullName.substring(k + 1);
k = firstName.indexOf(BLANK);
firstName = firstName.substring(0, k);
- III int firstBlank = fullName.indexOf(BLANK);
int secondBlank = fullName.indexOf(BLANK);
firstName = fullName.substring(firstBlank + 1, secondBlank + 1);

- (A) I only
(B) II only
(C) III only
(D) I and II only
(E) I, II, and III

2. A certain class, `SomeClass`, contains a method with the following header:

```
public int getValue(int n)
```

Suppose that methods with the following headers are now added to `SomeClass`:

I `public int getValue()`

II `public double getValue(int n)`

III `public int getValue(double n)`

Which of the above headers will cause an error?

- (A) None
 - (B) I only
 - (C) II only
 - (D) III only
 - (E) I and III only
3. Consider the following statement:

```
int num = /* expression */;
```

Which of the following replacements for `/* expression */` creates in `num` a random integer from 2 to 50, including 2 and 50?

- (A) `(int)(Math.random() * 50) - 2`
- (B) `(int)(Math.random() * 49) - 2`
- (C) `(int)(Math.random() * 49) + 2`
- (D) `(int)(Math.random() * 50) + 2`
- (E) `(int)(Math.random() * 48) + 2`

4. Consider the following code segment.

```
int num = 0, score = 10;  
if (num != 0 && score / num > SOME_CONSTANT)  
    statement1;  
else  
    statement2;
```

What is the result of executing this statement?

- (A) An `ArithmeticException` will be thrown.
- (B) A syntax error will occur.
- (C) `statement1`, but not `statement2`, will be executed.
- (D) `statement2`, but not `statement1`, will be executed.
- (E) Neither `statement1` nor `statement2` will be executed; control will pass to the first statement following the `if` statement.

1. What output is produced by the following line of code?

```
System.out.println("\"This is\n very strange\"");
```

- (A) \This is\n very strange\
- (B) "This is very strange"
- (C) This is
very strange
- (D) \"This is
very strange\"
- (E) "This is
very strange"

20. Consider the following segment of code,

```
String word = "conflagration";  
int x = word.indexOf("flag");  
String s = word.substring(0, x);
```

What will be the result of executing the above segment?

- (A) A syntax error will occur.
- (B) String s will be the empty string.
- (C) String s will contain "flag".
- (D) String s will contain "conf".
- (E) String s will contain "con".

34. When an integer is represented in base 16 (hexadecimal), the digits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F are used, where A-F represent the numbers 10-15. If base 16 is represented with the subscript _{hex} and base 10 is represented with the subscript _{dec}, then the decimal number 196 could be represented in hexadecimal as shown below:

$$196_{\text{dec}} = C4_{\text{hex}}$$

Which of the following is equal to $2AF_{\text{hex}}$?

- (A) 27_{dec}
- (B) 300_{dec}
- (C) 687_{dec}
- (D) 4002_{dec}
- (E) 6896_{dec}

2) Consider the incomplete method below. The method `square()` is intended to return the square of number passed to it.

```
public int square(int x)
{
    // missing code
}
```

Which of the code segments shown below can be used to replace `// missing code` so that `square()` will work as intended?

- (A) `return x * x;`
- (B) `return x ^ 2;`
- (C) `return x * 2;`
- (D) All of the above
- (E) None of the above

5) Consider the following method.

```
public void checkMarks(int x)
{
    if (x <= 40)
    {
        System.out.println("You have failed ");
    }
    else if (x >= 60)
    {
        System.out.println("You have passed ");
    }
    else
    {
        System.out.println("We are sorry ");
    }
}
```

What is printed as a result of the call `checkMarks(50)`?

- (A) We are sorry
- (B) You have passed
- (C) You have failed
- (D) You have failed We are sorry
- (E) None of these

6) The Java compiler _____.

- (A) Translates Java source code to byte code
- (B) Creates classes
- (C) Creates executable files
- (D) Produces the Java interpreter
- (E) All of above

7) Consider the following code segment.

```
String str = new String("Zeshan");  
System.out.println(str.length());
```

What is the output of the above code?

- (A) 0
- (B) 7
- (C) 6
- (D) 5
- (E) None of the above

8) Consider the following code segment.

```
int s = 0;  
int j = 0;  
int f = 3;  
int temp = (j * 4) + (f * f);  
System.out.println(temp);
```

What is the output of above code segment?

- (A) 9
- (B) 0
- (C) 3
- (D) 6
- (E) None of the above

9) Consider the following code segment.

```
System.out.printf("%s\n%s\n%s\n", "*", "***", "*****");
```

What is the output of above code segment?

(A) * *** *****

(B) *

(C) *

(D) *****

*

(E) *

10) Every class declaration must begin with which of the following keywords?

- (A) public
- (B) static
- (C) protected
- (D) private
- (E) None of above

11) Consider the following method grade().

```
public void grade(int studentGrade)
{
    if (studentGrade >= 90)
    {
        System.out.println("A");
    }
    else if (studentGrade >= 80)
    {
        System.out.println("B");
    }
    else if (studentGrade >= 70)
    {
        System.out.println("C");
    }
    else if (studentGrade >= 60)
    {
        System.out.println("D");
    }
    else
    {
        System.out.println("F");
    }
}
```

What is printed as a result of the call grade(59)?

- (A) A
- (B) F
- (C) D
- (D) B
- (E) None of the above

19) Consider the following code segment.

```
int x = 5;  
int y = 2;  
System.out.println(x / y - (double) (x / y));
```

What will be printed after running this code?

- (A) 0
- (B) 1
- (C) 0.5
- (D) 5.5
- (E) None of the above

12) Assume that variables x and y are of type double.

$!(x < y) \ \&\& \ !(x > y)$

The expression above is equivalent to which of the following?

- (A) true
- (B) false
- (C) $x == y$
- (D) $x != y$
- (E) $!(x < y) \ \&\& \ (x > y)$

20) Consider the following code segment.

```
if (x > 2)
{
    x = x * 2;
}
else if (x > 4)
{
    x = 0;
}
```

Which of the following code segments is equivalent to the code above?

I. `x + 4;`

II.

```
if (x > 2)
{
    x = x * 2;
}
```

III.

```
if (x > 4)
{
    x = 0;
}
```

(A) I only

(B) II only

(C) III only

(D) I, II and III

(E) None of the above

1. Which of the following pairs of declarations will cause an error message?

I double x = 14.7;
int y = x;

II double x = 14.7;
int y = (int) x;

III int x = 14;
double y = x;

- (A) None
- (B) I only
- (C) II only
- (D) III only
- (E) I and III only

2. What output will be produced by

```
System.out.print("\\* This is not\n a comment *\\");
```

- (A) * This is not a comment *
- (B) * This is not a comment *\
- (C) * This is not
a comment *
- (D) * This is not
a comment *\\
- (E) * This is not
a comment *\

3. Consider the following code segment

```
if (n != 0 && x / n > 100)  
    statement1;  
else  
    statement2;
```

If n is of type int and has a value of 0 when the segment is executed, what will happen?

- (A) An ArithmeticException will be thrown.
- (B) A syntax error will occur.
- (C) *statement1*, but not *statement2*, will be executed.
- (D) *statement2*, but not *statement1*, will be executed.
- (E) Neither *statement1* nor *statement2* will be executed; control will pass to the first statement following the if statement.

4. Refer to the following code fragment:

```
double answer = 13 / 5;  
System.out.println("13 / 5 = " + answer);
```

The output is

13 / 5 = 2.0

The programmer intends the output to be

13 / 5 = 2.6

Which of the following replacements for the first line of code will *not* fix the problem?

- (A) `double answer = (double) 13 / 5;`
- (B) `double answer = 13 / (double) 5;`
- (C) `double answer = 13.0 / 5;`
- (D) `double answer = 13 / 5.0;`
- (E) `double answer = (double) (13 / 5);`

5. What value is stored in result if

```
int result = 13 - 3 * 6 / 4 % 3;
```

- (A) -5
 - (B) 0
 - (C) 13
 - (D) -1
 - (E) 12
6. Suppose that addition and subtraction had higher precedence than multiplication and division. Then the expression

$2 + 3 * 12 / 7 - 4 + 8$

would evaluate to which of the following?

- (A) 11
 - (B) 12
 - (C) 5
 - (D) 9
 - (E) -4
7. Which is true of the following boolean expression, given that x is a variable of type double?

```
3.0 == x * (3.0 / x)
```

- (A) It will always evaluate to false.
- (B) It may evaluate to false for some values of x.
- (C) It will evaluate to false only when x is zero.
- (D) It will evaluate to false only when x is very large or very close to zero.
- (E) It will always evaluate to true.

8. Let x be a variable of type `double` that is positive. A program contains the boolean expression `(Math.pow(x,0.5) == Math.sqrt(x))`. Even though $x^{1/2}$ is mathematically equivalent to \sqrt{x} , the above expression returns the value `false` in a student's program. Which of the following is the most likely reason?
- (A) `Math.pow` returns an `int`, while `Math.sqrt` returns a `double`.
 - (B) x was imprecisely calculated in a previous program statement.
 - (C) The computer stores floating-point numbers with 32-bit words.
 - (D) There is round-off error in calculating the `pow` and `sqrt` functions.
 - (E) There is overflow error in calculating the `pow` function.

9. What will the output be for the following poorly formatted program segment, if the input value for `num` is 22?

```
int num = call to a method that reads an integer;
if (num > 0)
if (num % 5 == 0)
System.out.println(num);
else System.out.println(num + " is negative");
```

- (A) 22
- (B) 4
- (C) 2 is negative
- (D) 22 is negative
- (E) Nothing will be output.

10. What values are stored in x and y after execution of the following program segment?

```
int x = 30, y = 40;
if (x >= 0)
{
    if (x <= 100)
    {
        y = x * 3;
        if (y < 50)
            x /= 10;
    }
    else
        y = x * 2;
}
else
    y = -x;
```

- (A) $x = 30$ $y = 90$
- (B) $x = 30$ $y = -30$
- (C) $x = 30$ $y = 60$
- (D) $x = 3$ $y = -3$
- (E) $x = 30$ $y = 40$

1) Which of the following class names is incorrect?

- (A) APcs_2015_java
- (B) Apcs_java_2015
- (C) Java_APcs_2015
- (D) 2015_java_APcs

2) Which of the following comments is incorrect?

- (A) `///AP computer science java.`
- (B) `// AP computer science java.`
- (C) `///AP computer science.`
- (D) `/AP computer science/`

3) Which of the following comments is correct?

- (A) `/*ABCD*/`
 - (B) `*/ABCD*/`
 - (C) `/*ABCD/*`
 - (D) `**ABCD//`
-

4) Which of the following HelloWorld programs is correct?

(A)

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }
}
```

(B)

```
public class HelloWorld
{
    public static void main(String[] args)
    [
        System.out.println("Hello world");
    ]
}
```

(C)

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }
}
```

(D)

```
public class HelloWorld
(
    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }
)
```

5) Do comments affect the functionality of the program?

- (A) Yes, only while executing the class
- (B) No, not at all
- (C) Yes, only while executing the main method
- (D) No, only the methods are affected

6) Which of the following main method declarations is correct?

- (A) `public static void main(String() args);`
- (B) `public static String main(String[] args);`
- (C) `public static void main(String arg);`
- (D) `public static void main(String[] args);`

7) Which of the following statement(s) is a valid variable declaration and assignment?

- (A) `int x = 21.7;`
- (B) `int x;`
`x = 21.7;`
- (C) `int x = (int) 21.7;`
- (D) `int x = (double) 21.7;`

8) What is the result of the following expression?

```
int result = 2 - 6 / 3 * 4 + 2 % 5;
```

- (A) 4
- (B) -4
- (C) 3
- (D) -3

10. *Top-down* program development methodology is best characterized by:

- (A) Defining a class's methods and class variables first and instance variables and constructors later
- (B) Coding classes and methods that implement higher-level tasks first, using temporary "stubs" for lower-level classes and methods
- (C) Having a project leader design the software and then divide the work among programmers
- (D) Assembling the program from reusable components
- (E) Designing data structures first and classes and objects later

14. Which of the following statements about Java's platform independence are true?

- I. The number of bytes used by an `int` variable is the same on any computer.
- II. Java source code is compiled into bytecodes, which may then be run on any computer that has a Java Virtual Machine installed.
- III. Overflow in arithmetic operations occurs at the same values regardless of the platform on which the Java program is running.

- (A) I only
- (B) II only
- (C) I and II
- (D) II and III
- (E) I, II, and III

15. Given two initialized `String` variables, `str1` and `str2`, which of the following conditions correctly tests whether the value of `str1` is greater than or equal to the value of `str2` (in lexicographical order)?

- (A) `str1.compareTo(str2) == true`
- (B) `str1.compareTo(str2) >= 0`
- (C) `str1 >= str2`
- (D) `str1.equals(str2) || str1.compareTo(str2) == 1`
- (E) `str1.length() > str2.length() || str1 >= str2`

23. What is the output from the following code segment?

```
double pi = 3.14159;
int r = 100;
int area = (int)(pi * Math.pow(r, 2));
System.out.println(area);
```

- (A) 30000
- (B) 31415
- (C) 31416
- (D) 314159
- (E) Depends on the particular computer system

2. Assuming that `x` and `y` are `int` variables, the expression

```
!(x > y && y <= 0)
```

is equivalent to which of the following?

- (A) `!(x <= y) || (y > 0)`
- (B) `x > y && y <= 0`
- (C) `x <= y || y > 0`
- (D) `x > y || y < 0`
- (E) `x <= y && y <= 0`

3. Which of the following could serve as a postcondition in the following method?

```
// precondition: amt represents a positive value in dollars
//                and cents (for example, 1.15 represents
//                one dollar and fifteen cents)
private int process(double amt)
{
    return (int)(amt * 100 + 0.5) % 100;
}
```

- (A) Returns the cent portion in `amt`.
- (B) Returns the number of whole dollars in `amt`.
- (C) Returns `amt` converted into cents.
- (D) Returns `amt` rounded to the nearest integer.
- (E) Returns `amt` truncated to the nearest integer.

7. Which of the following Boolean expressions properly implement a comparison for equality of two String objects `str1` and `str2` and evaluate to true if and only if `str1` and `str2` hold the same values?

- I. `str1 == str2`
- II. `str1.equals(str2)`
- III. `str1.compareTo(str2) == 0`

- (A) I only
- (B) II only
- (C) I and II
- (D) II and III
- (E) I, II, and III

8. What is the output of the following code segment?

```
int a = 3;
int b = 4;
int c = 0;

if (a == b && b/c == 1)
{
    c = a * b;
}
else
{
    c = a + b * c;
    System.out.println(c);
}
```

- (A) Run-time division-by-zero error
- (B) 0
- (C) 3
- (D) 6
- (E) 12

14. Which of the following recommendations for testing software is NOT good advice?

- (A) Test a program with all possible values of input data.
- (B) When testing a large program, test the smaller pieces individually before testing the entire program.
- (C) If possible, use automated testing procedures or read test data from files so that you can re-run the tests after corrections have been made.
- (D) Design test data that exercises as many different paths through the code as is practical.
- (E) Test on data that is at the boundary of program conditionals to check for "off by one" errors.

18. A common use of hexadecimal numerals is to specify colors on web pages. Every color has a red, green, and blue component. In decimal notation, these are denoted with an ordered triple (x, y, z) , where x , y , and z are the three components, each an int from 0 to 255. For example, a certain shade of red, whose red, green, and blue components are 238, 9, and 63, is represented as $(238, 9, 63)$.

In hexadecimal, a color is represented in the format #RRGGBB, where RR, GG, and BB are hex values for the red, green, and blue. Using this notation, the color $(238, 9, 63)$ would be coded as #EE093F.

Which of the following hex codes represents the color $(14, 20, 255)$?

- (A) #1418FE
- (B) #0E20FE
- (C) #0E14FF
- (D) #0FE5FE
- (E) #0D14FF

19. In Java, a variable of type `int` is represented internally as a 32-bit signed integer. Suppose that one bit stores the sign, and the other 31 bits store the magnitude of the number in base 2. In this scheme, what is the largest value that can be stored as type `int`?

- (A) 2^{32}
- (B) $2^{32} - 1$
- (C) 2^{31}
- (D) $2^{31} - 1$
- (E) 2^{30}

20. Consider this code segment:

```
int x = 10, y = 0;
while (x > 5)
{
    y = 3;
    while (y < x)
    {
        y *= 2;
        if (y % x == 1)
            y += x;
    }
    x -= 3;
}
System.out.println(x + " " + y);
```

What will be output after execution of this code segment?

- (A) 1 6
- (B) 7 12
- (C) -3 12
- (D) 4 12
- (E) -3 6

Questions 21 and 22 refer to the following method, `checkNumber`, which checks the validity of its four-digit integer parameter.

```
/** @param n a 4-digit integer
 * @return true if n is valid, false otherwise
 */
boolean checkNumber(int n)
{
    int d1,d2,d3,checkDigit,nRemaining,rem;
    //strip off digits
    checkDigit = n % 10;
    nRemaining = n / 10;
    d3 = nRemaining % 10;
    nRemaining /= 10;
    d2 = nRemaining % 10;
    nRemaining /= 10;
    d1 = nRemaining % 10;
    //check validity
    rem = (d1 + d2 + d3) % 7;
    return rem == checkDigit;
}
```

A program invokes method `checkNumber` with the statement

```
boolean valid = checkNumber(num);
```

21. Which of the following values of `num` will result in `valid` having a value of `true`?
- (A) 6143
 - (B) 6144
 - (C) 6145
 - (D) 6146
 - (E) 6147
22. What is the purpose of the local variable `nRemaining`?
- (A) It is not possible to separate `n` into digits without the help of a temporary variable.
 - (B) `nRemaining` prevents the parameter `num` from being altered.
 - (C) `nRemaining` enhances the readability of the algorithm.
 - (D) On exiting the method, the value of `nRemaining` may be reused.
 - (E) `nRemaining` is needed as the left-hand side operand for integer division.

23. What output will be produced by this code segment? (Ignore spacing.)

```
for (int i = 5; i >= 1; i--)  
{  
    for (int j = i; j >= 1; j--)  
        System.out.print(2 * j - 1);  
    System.out.println();  
}
```

(A) 9 7 5 3 1
9 7 5 3
9 7 5
9 7
9

(B) 9 7 5 3 1
7 5 3 1
5 3 1
3 1
1

(C) 9 7 5 3 1
7 5 3 1 -1
5 3 1 -1 -3
3 1 -1 -3 -5
1 -1 -3 -5 -7

(D) 1
1 3
1 3 5
1 3 5 7
1 3 5 7 9

(E) 1 3 5 7 9
1 3 5 7
1 3 5
1 3
1

24. Which of the following program fragments will produce this output? (Ignore spacing.)

```
2 - - - -  
- 4 - - - -  
- - 6 - - -  
- - - 8 - -  
- - - - 10 -  
- - - - - 12
```

```
I for (int i = 1; i <= 6; i++)  
{  
    for (int k = 1; k <= 6; k++)  
        if (k == i)  
            System.out.print(2 * k);  
        else  
            System.out.print("-");  
    System.out.println();  
}
```

```
II for (int i = 1; i <= 6; i++)  
{  
    for (int k = 1; k <= i - 1; k++)  
        System.out.print("-");  
    System.out.print(2 * i);  
    for (int k = 1; k <= 6 - i; k++)  
        System.out.print("-");  
    System.out.println();  
}
```

```
III for (int i = 1; i <= 6; i++)  
{  
    for (int k = 1; k <= i - 1; k++)  
        System.out.print("-");  
    System.out.print(2 * i);  
    for (int k = i + 1; k <= 6; k++)  
        System.out.print("-");  
    System.out.println();  
}
```

- (A) I only
- (B) II only
- (C) III only
- (D) I and II only
- (E) I, II, and III

25: Consider this program segment:

```
int newNum = 0, temp;
int num = k;          //k is some predefined integer value  $\geq 0$ 
while (num > 10)
{
    temp = num % 10;
    num /= 10;
    newNum = newNum * 10 + temp;
}
System.out.print(newNum);
```

Which is a true statement about the segment?

- I If $100 \leq \text{num} \leq 1000$ initially, the final value of newNum must be in the range $10 \leq \text{newNum} \leq 100$.
- II There is no initial value of num that will cause an infinite while loop.
- III If $\text{num} \leq 10$ initially, newNum will have a final value of 0.

- (A) I only
- (B) II only
- (C) III only
- (D) II and III only
- (E) I, II, and III

26. Consider the method reverse:

```
/** Precondition: n > 0.  
 * Postcondition:  
 * - Returns n with its digits reversed.  
 * - Example: If n = 234, method reverse returns 432.  
 * @param n a positive integer  
 * @return n with its digits reversed  
 */  
int reverse(int n)  
{  
    int rem, revNum = 0;  
  
    /* code segment */  
  
    return revNum;  
}
```

Which of the following replacements for */* code segment */* would cause the method to work as intended?

```
I for (int i = 0; i <= n; i++)  
{  
    rem = n % 10;  
    revNum = revNum * 10 + rem;  
    n /= 10;  
}
```

```
II while (n != 0)  
{  
    rem = n % 10;  
    revNum = revNum * 10 + rem;  
    n /= 10;  
}
```

```
III for (int i = n; i != 0; i /= 10)  
{  
    rem = i % 10;  
    revNum = revNum * 10 + rem;  
}
```

- (A) I only
- (B) II only
- (C) I and II only
- (D) II and III only
- (E) I and III only

14) What will be displayed upon execution of this code segment?

```
Double i = new Double(257.578);  
int x = i.intValue();  
System.out.print(x);
```

(A) 0

(B) 1

(C) 258

(D) 257

15) Which of these methods of the String class can be used to test to strings for equality?

(A) isequal()

(B) isequals()

(C) equal()

(D) equals()

16) What will be displayed upon execution of this code segment?

```
String obj = "I" + "like" + "Java";  
System.out.println(obj);
```

(A) I

(B) Like

(C) Java

(D) IlikeJava

17) What will be displayed upon execution of this code segment?

```
String obj = "I LIKE JAVA";  
System.out.println(obj.length());
```

- (A) 9
- (B) 10
- (C) 11
- (D) 12

18) What will be displayed upon execution of this code segment?

```
String obj = "hello";  
String obj1 = "world";  
String obj2 = "hello";  
System.out.println(obj.equals(obj1) + " " + obj.equals(obj2));
```

- (A) false false
- (B) true true
- (C) true false
- (D) false true

19) Which of these methods of the String class is used to extract a substring from a String object?

- (A) substring()
- (B) Substring()
- (C) SubString()
- (D) None of the above

20) What will be displayed upon execution of this code segment?

```
String s1 = "Hello World";  
String s2 = s1.substring(0 , 4);  
System.out.println(s2);
```

- (A) Hell
- (B) Hello
- (C) Worl
- (D) World

21) What will be displayed upon execution of this code segment?

```
String s = "Hello World";  
int i = s.indexOf('o');  
int j = s.lastIndexOf('l');  
System.out.print(i + " " + j);
```

- (A) 4 8
- (B) 5 9
- (C) 4 9
- (D) 5 8

22) What will be displayed upon execution of this code segment?

```
String s1 = "Moksh";  
String s2 = "Moksh";  
String s3 = "Jawa";  
System.out.print(s1.compareTo(s2) + " ");  
System.out.print(s1.compareTo(s3) + " ");  
System.out.print(s3.compareTo(s1));
```

- (A) 0 0 0
- (B) 0 3 3
- (C) 0 -3 -3
- (D) 0 3 -3