1. What does “short circuit evaluation” mean for the | | operator?
2. if the left operand evaluates to false, then the right operand is not evaluated
3. if the left operand evaluates to true, then the right operand is not evaluated
4. if the right operand evaluates to false, then the left operand is not evaluated
5. if the right operand evaluates to true, then the left operand is not evaluated
6. What is the value of mins after the following poorly formatted code fragment is executed?

int mins = 100;

if ( mins < 60 )

if ( mins >= 10)

mins = 10;

else

mins = mins % 60;

1. 1
2. 10
3. 40
4. 100
5. Consider the code below. Determine how many stars are printed.

for (int i = 0; i < 5; i ++)

 for (int j = i ; j < 5; j++)

 System.out.println(“\*”);

1. 5
2. 10
3. 15
4. 20
5. The order of operations for the Boolean operators and (&&), or (| |), and not ( ! ) is:
6. and, or, not
7. not, and, or
8. or, and not
9. all three have the same priority – they are evaluated left to right

13. public static int result(int m, int n)

{

 int j, temp;

 temp = 1;

 for ( j = 1; j <= n; j++)

 temp \*= m;

 return temp;

}

Of the following, which best indicates the task performed by the method above if only positive parameters are passed to the method?

1. it approximates the mth root of n
2. it approximates the nth root of m
3. it computes the mth power of n
4. it computes the nth power of m
5. it computes the roots of a quadratic with positive coefficients and with leading coefficient 1

14. Consider the method *total* below.

public int *total* (int result, int a, int b) {

 **if** (a==0){

 **if** (b==0)

 **return** result \* 2;

 }

 **else**

 **return** result \* 3;

}

The assignment statement

x = *total* (5, 0, 1);

must result in

1. x being assigned the value 0
2. x being assigned the value 5
3. x being assigned the value 10
4. x being assigned the value 15
5. none of the above
6. Consider the following code segment.

int k = 5;

while(k >= 0 && k % 2 == 1)

{

 System.out.print(k);

 k = 2\*k – 5;

}

What is printed as a result of executing this code segment?

A. 5

B. 5 1

C. 5 3

D. 5 3 1

E. An unending sequence of 5's is printed since the loop never exits

1. The following code segment is intended to sum the first 10 positive odd integers.

sum = 0;

for(int k = 1; k <= 10; k++)

{

 sum += 2\*k + 1;

}

Which of the following best describes the error, if any, in this code.

A. The segment works as intended.

B. The segment sums the first 9 odd integers

C. The segment sums the first 20 odd integers

D. The segment leaves out the first odd integer and includes the eleventh odd integer in the sum.

E. The variable sum is incorrectly initialized. The segment would work as intended if sum were initialized to 1.

1. Consider the following code segment.

for(int num = 5; num > 0; num--)

{

 if(num % 2 = 1)

 {

 for(int star = 0; star < num; star++)

 System.out.print("\*");

 System.out.println();

 }

}

What will be printed when this code segment is executed?

A. \*

 \*\*

 \*\*\*

 \*\*\*\*

 \*\*\*\*\*

B. \*

 \*\*\*

 \*\*\*\*\*

C. \*\*\*\*\*

 \*\*\*\*

 \*\*\*

 \*\*

 \*

D. \*\*\*\*\*

 \*\*\*

 \*

E. \*\*\*\*

 \*\*

1. Consider the following static method

public static double getSomething(int val)

{

 val = 2 + val;

 val = val + 3\*val;

 return val;

}

Which of the following could be used to replace the body of getSomething so that the modified version will return the same result as the original version for all values of the parameter val.

A. return 4\*val + 2;

B. return 4\*val + 6;

C. return 4\*val + 8;

D. return 7\*val + 6;

E. return 7\*val + 8;

1. Consider the following code segment.

int value = 7;

for(int modulus = 6; value % modulus != 0 && modulus != 0; modulus = modulus - 2)

{

 value += 3;

}

System.out.println( value+ “ “ + modulus);

What is printed when this code is executed?

A. 16 0

B. 13 2

C. 10 4

D. 7 6

E. Nothing is printed because a division-by-zero will generate an ArithmeticException

1. Consider the following code segment.

double val = 0.5;

int count = 0;

while(val > 0.1)

{

 val = val \* val;

 count ++;

}

What are the values of val and count when the loop exits?

1. val = 0.0625, count = 2
2. val = 0.0625, count = 3
3. val = 0.03125, count = 3
4. val = 0.03125, count = 4
5. The loop never exits.
6. Suppose x is an int variable that holds a positive integer. Consider the following five expressions:

 a. x % 100 / 10

 b. x / 10 % 10

 c. (x - x % 100) / 10

 d. x / 10 - x / 100 \* 10

 e. (x - x / 100 \* 100) / 10

Which one of them produces a value different from the other four for some values of x?

(A) a.

(B) b.

(C) c.

(D) d.

(E) e.

22. Which of the following statements is equivalent to

 if (a == 0 || b\*b - 4\*a\*c <= 0)

 return false;

 else

 return true;

(A) return a == 0 || b\*b - 4\*a\*c <= 0;

(B) return a == 0 && b\*b - 4\*a\*c > 0;

(C) return a != 0 || b\*b - 4\*a\*c > 0;

(D) return a != 0 && b\*b - 4\*a\*c > 0;

(E) None of the above

1. If addition had higher precedence than multiplication, then the value of the expression

**1 + 2 \* 3 + 4 \* 5**

would be which of the following?

(A) 27

(B) 47

(C) 65

(D) 69

(E) 105

1. Consider the following syntax diagram.

 **string X X X string**

 **X X**

Which of the following is NOT a valid string according to this diagram?

(A) X

(B) XX

(C) XXXX

(D) XXXXX

(E) XXXXXXX

1. Consider the following program segment.

 int x, y;

 x = <read in the value of x> // integer is read

 **while** (x != 0) {

 y = <read in the value of y> // integer is read

 **if** (y == 0)

 **System.out.println(**“yes”);

 **else**

 **System.out.println(**“no”);

 x = <read in the value of x> // integer is read

 }

 return 0;

}

Which of the following input sequences ensures that every statement of the program is executed at least once?

1. 1 1 1 2 0
2. 1 0 1 1 0
3. 1 0 1 0 1 0 0
4. 1 0 0
5. 1 1 0

**Questions 4 - 5** refer to the following information,

The code

if ( n == 1 )

 k -= 1;

else if ( n == 2 )

 k -= 2;

is rewritten in the form

if ( <condition> )

<assignment statement>;

where <condition> and <assignment statement> are chosen so that the rewritten code performs the same task as the original code. Assume that both n and k are integer variables.

1. Which of the following could be used as <condition>?
2. (n==1) || (n==2)
3. (n==1) && (n==2)
4. (n >= 1) && (n <= 2)

(A) I only

(B) II only

(C) III only

(D) I and III only

(E) II and III only

1. Assume that evaluating <condition> changes neither n nor k. Which of the following could be used as <assignment statement>?
2. k -= n;
3. k -= 1;
4. k -= 2;
5. k += n;
6. k = n – k;
7. Consider the following code segments.

***Segment 1***

x = <read in the value of x>;

**while** ( x > 0 ) {

 **System.out.println(**x);

 x -= 1;

}

**while** ( x > 0 ) {

 **System.out.println(**x);

 x -= 1;

}

***Segment 2***

x =< read in the value of x>;

**while** ( x > 0 ) {

 **System.out.println(**x);

 x -= 1;

}

Under which of the following conditions will the two code segments produce exactly the same output?

1. The value read in the first line is less than zero.
2. The value read in the first line is greater than zero.
3. The value read in the first line is equal to zero.

(A) I only

(B) II only

(C) III only

(D) I and III only

(E) I, II, and III

1. The Boolean expression

(num > max) || !(max < num)

can be simplified to

(A) max != num

(B) max == num

(C) (num < max) && !(max < num)

(D) false

(E) true