***Flow Control***

**For more details on SUN Certifications, visit** [**JavaScjpDumps**](http://www.javascjpdumps.blogspot.com/)

**Q: 01 Given:**

**10. public class Bar {**

**11. static void foo( int... x ) {**

**12. // insert code here**

**13. }**

**14. }**

**Which two code fragments, inserted independently at line 12, will allow the class to compile? (Choose**

**two.)**

A. foreach( x ) System.out.println(z);

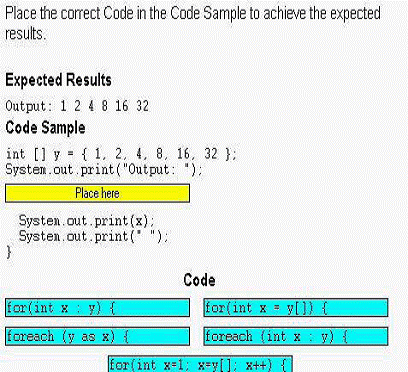
B. for( int z : x ) System.out.println(z);

C. while( x.hasNext() ) System.out.println( x.next() );

D. for( int i=0; i< x.length; i++ ) System.out.println(x[i]);

**ANSWER**=B,D

**Q: 02 Click the Task button.**



**Solution:**

int [ ] y={1,2,4,8,16,32};

System.out.print("output : ");

**for(int x : y ) {**

System.out.println(x);

System.out.println(" ");

**Q: 03 Given:**

**25. int x = 12;**

**26. while (x < 10) {**

**27. x--;**

**28. }**

**29. System.out.print(x);**

**What is the result?**

A. 0

B. 10

C. 12

D. Line 29 will never be reached.

**ANSWER**=C

**Q: 04 Given:**

**11. public static void main(String[] args) {**

**12. Object obj = new int[] { 1, 2, 3 };**

**13. int[] someArray = (int[])obj;**

**14. for (int i : someArray) System.out.print(i + " ");**

**15. }**

**What is the result?**

A. 1 2 3

B. Compilation fails because of an error in line 12.

C. Compilation fails because of an error in line 13.

D. Compilation fails because of an error in line 14.

E. A ClassCastException is thrown at runtime.

**ANSWER**=A

**Q: 05 Given:**

**11. public static void main(String[] args) {**

**12. for (int i = 0; i <= 10; i++) {**

**13. if (i > 6) break;**

**14. }**

**15. System.out.println(i);**

**16. }**

**What is the result?**

A. 6

B. 7

C. 10

D. 11

E. Compilation fails.

F. An exception is thrown at runtime.

**ANSWER**=E

**Q: 06 Given:**

**11. public static void main(String[] args) {**

**12. Integer i = new Integer(1) + new Integer(2);**

**13. switch(i) {**

**14. case 3: System.out.println("three"); break;**

**15. default: System.out.println("other"); break;**

**16. }**

**17. }**

**What is the result?**

A. three

B. other

C. An exception is thrown at runtime.

D. Compilation fails because of an error on line 12.

E. Compilation fails because of an error on line 13.

F. Compilation fails because of an error on line 15.

**ANSWER**=A

**Q: 07 Given:**

**10. public class ClassA {**

**11. public void count(int i) {**

**12. count(++i);**

**13. }**

**14. }**

**And:**

**20. ClassA a = new ClassA();**

**21. a.count(3);**

**Which exception or error should be thrown by the virtual machine?**

A. StackOverflowError

B. NullPointerException

C. NumberFormatException

D. IllegalArgumentException

E. ExceptionInInitializerError

**ANSWER**=A

**Q: 08 Given:**

**35. int x = 10;**

**36. do { 37. x--;**

**38. } while (x < 10);**

**How many times will line 37 be executed?**

A. ten times

B. zero times

C. one to nine times

D. more than ten times

**ANSWER**=D

**9. Given the following code:**

**public class OrtegorumFunction {**

**public int computeDiscontinuous(int x) {**

**int r = 1;**

**r += x;**

**if ((x > 4) && (x < 10)) {**

**r += 2 \* x;**

**} else (x <= 4) {**

**r += 3 \* x;**

**} else {**

**r += 4 \* x;**

**}**

**r += 5 \* x;**

**return r;**

**}**

**public static void main(String [] args) {**

**OrtegorumFunction o = new OrtegorumFunction();**

**System.out.println("OF(11) is: " + o.computeDiscontinuous(11));**

**} }**

**What is the result?**

A. OF(11) is: 45

B. OF(11) is: 56

C. OF(11) is: 89

D. OF(11) is: 111

E. Compilation fails.

F. An exception is thrown at runtime.

**ANSWER**=**E**is correct. The if statement is illegal. The if-else-else must be changed to if-else  
if-else, which would result in OF(11) is: 111.

**10. Given:**

**1. class Crivitch {**

**2. public static void main(String [] args) {**

**3. int x = 0;**

**4. // insert code here**

**5. do { } while (x++ < y);**

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

**7. }**

**8. }**

**Which, inserted at line 4, produces the output 12?**

A. int y = x;

B. int y = 10;

C. int y = 11;

D. int y = 12;

E. int y = 13;

F. None of the above will allow compilation to succeed.

**ANSWER**= **C**is correct. x reaches the value of 11, at which point the while test fails.x is then incremented (after the comparison test!), and the println() method runs.

**11. Given:**

**class Swill {**

**public static void main(String[] args) {**

**String s = "-";**

**switch(TimeZone.CST) {**

**case EST: s += "e";**

**case CST: s += "c";**

**case MST: s += "m";**

**default: s += "X";**

**case PST: s += "p";**

**}**

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

**}**

**}**

**enum TimeZone {EST, CST, MST, PST }**

**What is the result?**

A. -c

B. -X

C. -cm

D. -cmp

E. -cmXp

F. Compilation fails.

G. An exception is thrown at runtime.

**ANSWER**= **E**is correct. It’s legal to use enums in a switch, and normal switch fall-through logic applies; i.e., once a match is made the switch has been entered, and all remaining blocks will run if no break statement is encountered.

**12. Given:**

**class Circus {**

**public static void main(String[] args) {**

**int x = 9;**

**int y = 6;**

**for(int z = 0; z < 6; z++, y--) {**

**if(x > 2) x--;**

**label:**

**if(x > 5) {**

**System.out.print(x + " ");**

**--x;**

**continue label;**

**}**

**x--;**

**}**

**}**

**}**

**What is the result?**

A. 8

B. 8 7

C. 8 7 6

D. Compilation fails.

E. An exception is thrown at runtime.

**ANSWER**=**D**is correct. A labeled continue works only with loops. In this case, although the label is  
legal, label is not a label on a loop statement, it’s a label on an if statement.