

1. ข้อใดให้ผลลัพธ์เป็น true

```
int guests = 100 ;
```

```
int tables = 20 ;
```

- a. `guests > tables*5`
- b. `guests/5 - tables >= 0`
- c. `guests-10 == 4*tables+20`
- d. `!(tables+4 > guests/5)`

2. ข้อใดแสดงค่าของตัวแปรของ face

```
num = 7 ;
```

```
face = num > 5 ? "Head" : "Tail" ;
```

```
System.out.println(face);
```

3. คำสั่ง switch (expression) expression ในข้อใดใช้ได้กับคำสั่ง switch

```
int month = 2 ;
```

- a. `month == 2`
- b. `month%2`
- c. `month/2.0`
- d. `Math.sqrt(month)`

4. ข้อใดถูกต้องในการประกาศตัวแปร (เลือก 2 ข้อ)

```
int num = 5 ;
```

- a. `boolean isOdd = num%2 ;` ✗
- b. `boolean isOdd = num%2 != 0 ;`
- c. `boolean isOdd = num = 1 ;`
- d. `boolean isOdd = num%2 == 1 ;`

5. ข้อใดแสดงผลลัพธ์การดำเนินการของชุดคำสั่งต่อไปนี้

```
int menu= 3 ;
```

```
if (menu == 1 && menu == 2 && menu == 3){
```

```
    System.out.println("Please insert money more.");
```

```

    } else if (menu > 3 && menu < 6 ) {
        System.out.println("Give a product.");
    } else {
        System.out.println("Cancel the product.");
    }
}

```

6. คำสั่ง switch (expression) ชนิดข้อมูล (data type) ของผลลัพธ์ของ expression ในข้อใดใช้ไม่ได้กับ คำสั่ง switch

- a. int
- b. char
- c. boolean
- d. short

7. ข้อใดไม่ถูกต้องเกี่ยวกับการใช้คำสั่ง switch ใน Java

- a. ในแต่ละ case เป็นการเปรียบเทียบแบบเท่ากัน (equality) โดยอัตโนมัติ
- b. ในแต่ละ case เมื่อเจอคำสั่ง break จะหยุดการทำงานที่เหลือของ switch และจะไปทำคำสั่ง ถัดไปจากคำสั่ง switch
- c. ใน case ที่เป็น default ไม่ต้องใช้คำสั่ง break
- d. ใช้ ; เพื่อเชื่อมหลาย ๆ case ที่ต้องการทำคำสั่ง statement เหมือน ๆ กันเช่น case 'a' ; 'b' ; 'c' : System.out.println("Great job!") ;

8. บรรทัดใดที่เป็นสาเหตุทำให้โปรแกรม compile ไม่ผ่าน

- 1: int face = 5 ;
- 2: int guess = 2 ;
- 3: if (guess !< face)
- 4: System.out.println("Greater or equal!") ;
- 5: else
- 6: System.out.println("Less than!") ;
- 7: System.out.println("Please guess again.") ;

9. ข้อใดถูกต้องเกี่ยวกับการใช้คำสั่ง switch ใน Java
- a. ในแต่ละ case สามารถใช้ relational operator ได้
 - b. ในแต่ละ case ต้องมีอย่างน้อย 1 คำสั่ง (statement)
 - c. ในแต่ละ case สามารถใส่คำสั่ง break หรือไม่ใส่ (optional)
 - d. ต้องมี default ทุกครั้งที่มีการใช้คำสั่ง switch

10. ข้อใดแสดงค่าของตัวแปรของ money

```
int menu = 4 ;  
money = menu ==1 || menu ==3 ? menu*5 : menu*10+5 ;  
System.out.println(money);
```

11. ข้อใดแสดงผลลัพธ์ของการดำเนินการของชุดคำสั่งต่อไปนี้

```
int menu = 2 ;  
if (menu >=1) {  
    System.out.println("You choose menu 1") ;  
}  
if (menu >= 2) {  
    System.out.println("You choose menu 2") ;  
}  
if (menu >=3 ) {  
    System.out.println("You choose menu 3") ;  
} else {  
    System.out.println("You do not select any menu.") ;  
}
```

12. ข้อใดเปรียบเทียบข้อมูลไม่ถูกต้อง

- a.
- ```
String a = new String("Java") ;
String b = new String("Java") ;
```

```
if (a.equals(b)) {
 System.out.println("both a and b have the same value.");
}
```

b.

```
char key = 'A' ;
if (key < 'Z'){
 System.out.println("Your key is less than 'Z'.");
}
```

c.

```
int result = 9 ;
boolean isValid = result > 5 ;
if (isValid == result%2 == 1) {
 System.out.println("Num is Odd.");
}
```

d.

```
double num = 1.0/3 ;
double num2 = 1.0-2.0/3 ;
if(num != num2) {
 System.out.println("Both num and num2 are not the same value.");
}
```

1. What is the value of *result* after the program is run?

```
int result = 0;
int i = 1;
while (i < 4) {
 result = result + 2 * i;
 i++;
}
```

2. What is the value of *result* after the program is run?

```
int result = 0, b = 5, i = 1;
do {
 result = result + i;
 i++;
} while (result <= b);
```

3. What is the value of *result* after the program is run?

```
int a = 0, b = 1, result = 0;
for (int i = 0; i <= 4; i++) {
 result = a + b;
 a = b;
 b = result;
}
```

4. What is the value of *a[0]* after the program is run?

```
double[] a = {3, 4, 3.5, 2.5, 3, 3.5};
double temp;
for (int i = 0; i < a.length - 1; i++) {
 for (int j = i + 1; j < a.length; j++) {
 if (a[i] < a[j]) {
 temp = a[i];
 a[i] = a[j];
 a[j] = temp;
 }
 }
}
```

5. What is the value of *someThing03* after calling the *someMethod()* as the given code?

```
private static double[] someMethod(double[] a) {
 double b = a[0];
 double c = a[0];
 for (double temp : a) {
 if (temp > b) {
 b = temp;
 }
 if (temp < c) {
 c = temp;
 }
 }
 return new double[]{b,c};
}
```

This is the code that call the *someMethod()* in main method.

```
double[] someThing02 = {3, 4, 3.5, 2.5, 3, 3.5};
double[] someThing03;
someThing03= someMethod(someThing02);
```

6-9 Fill in the blank using the given choices to complete the average calculation method on any array of type double.

```
public static double average(double[] grades) {
 double [6. _____];
 for ([7. _____]: grades) {
 result = [8. _____];
 }
 return result/[9. _____];
}
```

10. Fill in the blank using the given choices to complete the balance calculation. The return with the given interest are included each month. There are 10 months to calculate.

```
double interest = 0.07;
double balance = 5000;
int noOfMonth = 10;
for ([10. _____]) {
 balance += balance * interest / 12;
}
System.out.println(balance);
```

11-12 Fill in the blank using the given choices to complete the prime checking method. 3, 5, 7, 11, 13, 17,... are the prime number.

```
private static boolean isPrime(int num) {
 for ([11. _____]) {
 if ([12. _____]) {
 return false;
 }
 }
 return true;
}
```

1)

```
int x[] = new int[100] ;
```

จากส่วนของโปรแกรมที่กำหนด

ถ้า x มี L-Value อยู่ที่ ตำแหน่ง F0000 x[20] จะเป็นตำแหน่งใดในหน่วยความจำ (ตอบเป็นเลขฐาน 16)

2)

```
int x[] = new int[1024] ;
```

จากส่วนของโปรแกรมที่กำหนด

ถ้า x มี LValue อยู่ที่ ตำแหน่ง F0000 , x[1011] จะมีค่า R-Value เป็นเท่าใด

3)

```
String x[] = { "Somchai", "Somchart", "Somsri", "Somying", "Somnamnha" } ;
```

```
String y[] = new String[x.length];
```

```
String z[] = new String[x.length];
```

```
System.arraycopy(x, 0, y, 0, x.length);
```

```
System.arraycopy(z, 0, z, 0, z.length);
```

จากส่วนของโปรแกรมที่กำหนด จะมี object (instance) เกิดขึ้นกี่ objects

What is the output of the following code fragment:

```
int[] zip = new int[5];
zip[0] = 7;
zip[1] = 3;
zip[2] = 4;
zip[3] = 1;
zip[4] = 9;
System.out.println(zip[2 + 1]);
```

How many objects are present after the following code fragment has executed?

```
double[] ann = new double[7];
```

```
double[] bob;
bob = ann;
```

What is the output of the following code fragment:

```
int[] zip = new int[5];
zip[0] = 7;
zip[1] = 3;
zip[2] = 4;
zip[3] = 1;
zip[4] = 9;
int j = zip[1];
System.out.println(zip[j+1]);
```

Fill in the blanks of the following code fragment so that the elements of the array are printed in **reverse order**, starting with the last element.

```
int[] egArray = { 2, 4, 6, 8, 10, 1, 3, 5, 7, 9 };

for (int _____ ; _____ ; _____)
 System.out.print(egArray[index] + " ");
```



What does the following method do?

```
void spread (int[] values){
 for (int index= 1 ; index < values.length ; index++)
 values[index] = values[0];
}
```

What is the output of the following program fragment:

```
int[] array = { 1, 4, 3, 6, 8, 2, 5};
int what = array[0];

for (int index=0; index < array.length; index++) {
 if (array[index] > what)
 what = array[index];
}
System.out.println(what);
```

What is the output of the following program fragment:

```
class ChangeIt {
 static void doIt(int[] z) {
 z[0] = 0;
 }
}

class TestIt {
 public static void main (String[] args) {
 int[] myArray = {1, 2, 3, 4, 5} ;

 ChangeIt.doIt(myArray);

 for (int j=0; j<myArray.length; j++)
 System.out.print(myArray[j] + " ") ;
 }
}
```

What is the output of the following?

```
class LowHighSwap {
 static void doIt(int[] z) {
 int temp = z[z.length-1] ;
 z[z.length-1] = z[0] ;
 z[0] = temp;
 }
}

class TestIt {
 public static void main (String[] args) {
 int[] myArray = {1, 2, 3, 4, 5} ;

 LowHighSwap.doIt(myArray);

 for (int j=0; j<myArray.length; j++)
 System.out.print(myArray[j] + " ") ;
 }
}
```