

ICSE Computer Applications Class 10

These questions will help you revise for Section A of Computer Applications Class 10 board examination. **Solutions are provided after the questions.**

Answer the following questions

1. Name the four basic principles of Object Oriented Programming
2. Define the following:
 - i) Abstraction
 - ii) Encapsulation
3. What are constants in Java? Give an example.
4. Name the wrapper classes of char type and boolean type.
5. Why an object is called an instance of a class? Explain.
6. Name the two types of constructors.
7. State the method that:
 - i) Converts a String to primitive float type.
 - ii) Determines if the specified character is an uppercase character.
8. How is implicit conversion different from explicit conversion? Give an example.
9. How are private members of a class different from public members?
10. What are the two ways of invoking methods?
11. List out the access specifiers in java?
12. Explain the terms operator precedence and associativity.
13. Name the package that contains:
 - i) Scanner class
 - ii) String class
14. Name the type of error (syntax, runtime or logical error) in each case given below:
 - i) `Math.sqrt(25 - 50)`
 - ii) `double x;y;z;`
15. Differentiate between the following
 - i) `System.exit(0)` and `break`
 - ii) Null loop and an infinite loop.
 - iii) `break` and `continue`
 - iv) Boxing-Unboxing
 - v) Searching and Sorting
 - vi) while loop and do-while loop
 - vii) Single Dimensional Array and Double Dimensional Array
 - viii) `length` and `length()`
 - ix) Linear search and Binary search
 - x) Subscript and subscripted variable
 - xi) Constructor and Method
 - xii) `next()` and `nextLine()`
 - xiii) `==` and `equals()`
 - xiv) Syntax Error and Logical Error
 - xv) Primitive Data Types and Composite Data Types

Evaluate the given expressions:

Question 1

```
int a=4, b=2, c=8;
i) a + b * c %a
ii) a + b / c + a
iii) a % b + b % c + b / a + c / b
iv) c += a++ - ++b + a - c
```

Question 2

```
int m=100, n=200, p=500;
i) m + n > n - p
ii) m + p % m == n - 100
iii) !(m * 2 == n)
iv) m < n && p % m == 0
v) n % m != p % p
```

Determine the output of the following statements

1. System.out.println (1+2*5%3);
2. System.out.println (1.5 + 2*'b');
3. char ch = 'X';
System.out.println(ch++ + 5);
System.out.println(--ch);
4. System.out.println(Math.pow (25,0.5));
5. System.out.println (Math.max('C','G'));
6. System.out.println ("ABCD" + 10 + 10.5);
7. int m=2, n=10;
System.out.println(m * 2 == n);
System.out.println(!(m<n && n>m));
8. System.out.println("LAPTOP".charAt(0) + "LENOVO".charAt(1));
9. System.out.println(Character.toUpperCase("CLOcK".charAt(3)));
10. System.out.println(Character.isLetter('j'));

Analyze how many times the following loops gets executed and write the output

Question 1

```
int x=100;
while (x!=0)
{
    System.out.println(x);
    if (x % 6 == 0)
        break;
    x--;
}
```

Question 2

```
int m=2, n=5;
while (m <= n)
{
    System.out.println(m + " " + n);
    m++; n++;
}
```

Question 3

```
int i = 0;
while (++i <= 10)
{
    if (i % 2 == 0)
        continue;
    System.out.println(i);
}
```

Question 4

```
int i;  
for (i = 50; i >= 10; i -= 10)  
System.out.println(i);  
System.out.println(++i*10);
```

Question 5

```
for (int i=3; i>=1; i--)  
{  
    for (int j=i; j >= 1; j--)  
        System.out.println(j);  
    System.out.println();  
}
```

Write the Output

Question 1

```
String s1="BASIC", s2="BASE";  
System.out.println(s1.length() + s2.length());  
System.out.println(s1.charAt(0) == s2.charAt(0));  
System.out.println(s2.charAt(s1.length() - 3));  
System.out.println(s1.startsWith(s2.substring(0,2)));  
System.out.println(s1.concat(" "+s2));  
System.out.println(s1.substring(0,1).concat(s2.substring(1)));  
System.out.println(s1.compareTo(s2));  
System.out.println(s1.toLowerCase().charAt(3));  
System.out.println(s2.substring(2, s2.length()));  
System.out.println(s1.replace(s1.charAt(s1.length() - 1), 'L'));
```

Question 2

```
int arr[] = {12, 10, 5, 8, 7};  
System.out.println(arr.length + arr[arr.length - 1]);  
System.out.println(arr[0]++ + --arr[2] * arr[4]);  
for (int i = 0; i < arr.length; i++)  
    System.out.print(arr[i] + " ");  
System.out.println("\n is the output");
```

Question 3

```
char x[] = {65, 66, 67, 68, 32}; //Note: 32 is the ASCII of space char  
System.out.println(x[0] + " " + x[3]);  
System.out.println(++x[1]);  
System.out.println(x[1]);  
for (int i = 0; i < x.length; i++)  
    System.out.println(x[i]);
```

Question 4

```
String st[] = {"MARS", "EARTH", "MOON", "MERCURY"};  
System.out.println(st.length);  
System.out.println(st[0].length() > st[2].length());  
System.out.println(st[1].indexOf('S'));  
System.out.println(st[3].charAt(0) == st[2].charAt(0));  
System.out.println(st[0].charAt(0) + " " + st[1].charAt(0));  
System.out.println(st[1].charAt(1) + st[2].charAt(0));  
System.out.println(st[2].length() + st[3].length());
```

Question 5

```
double x[]={1, 2, 3, 4.56}; int a=2;  
System.out.println(x[1]);  
System.out.println(Double.toString(x[2] + x[1]));  
System.out.println(x[0] + x[1] * x[2]);  
System.out.println(++x[++a]);
```

Question 6

```
String s="Examination";  
int a=20, b=19;  
System.out.println(s+a+b);  
System.out.println(a+b+s);  
System.out.println("A+B+" is the Output");  
System.out.println("Output is " + 'A'+'B');  
System.out.println("Output is " + ('A'+'B'));  
System.out.println("Output is " + a*b%2);  
System.out.println("Output is "+ 'a'+5);  
System.out.println('a' + 5 + "Output");
```

Rewrite as directed

Question 1

Using single if statement:

```
if (code == 'G')
System.out.println("Green");
if (code == 'g')
System.out.println("Green");
```

Question 2

Using while loop:

```
for (int x = 1, y = 2; x <= 5 && y <= 6; x++, y++)
System.out.println(x + "\n" + y);
```

Question 3

Using do...while loop:

```
for (int a = 125; a > 0; a = a / 10)
{
System.out.println(a % 10);
}
```

Question 4

Using ternary operator:

```
if (a % 2 == 0)
System.out.println("EVEN");
else
System.out.println("ODD");
```

Question 5

Using nested if statement:

```
if (a >= 10 && a <= 20)
System.out.println("Number in range");
else
System.out.println("Number not in range");
```

Solutions

Answer the following questions

Question 1

Name the four basic principles of Object Oriented Programming.

Abstraction
Encapsulation
Inheritance
Polymorphism

Question 2

Define the following:

i) Abstraction

Abstraction refers to the act of representing essential features without including the background details or explanations.

ii) Encapsulation

Wrapping of data and functions that operate on that data into a single unit is called Encapsulation.

Question 3

What are constants in Java? Give an example.

A variable whose value cannot change once it is assigned is termed as Constant. In Java, we declare a constant by using the keyword 'final' as shown in the below example:

```
final int DAYS_IN_A_WEEK = 7;
```

Question 4

Name the wrapper classes of char type and boolean type.

The wrapper class of char is Character and boolean is Boolean.

Question 5

Why an object is called an instance of a class? Explain.

A class can create objects of itself with different characteristics and common behaviour. So, we can say that an Object represents a specific state of the class. For these reasons, an Object is called an Instance of a Class.

Question 6

Name the two types of constructors.

Two types of constructors are Parameterised constructors and Non-Parameterised constructors.

Question 7

State the method that:

i) Converts a String to primitive float type.

Float.parseFloat()

ii) Determines if the specified character is an uppercase character.

Character.isUpperCase()

Question 8

How is implicit conversion different from explicit conversion? Give an example.

In an implicit conversion, the result of a mixed mode expression is obtained in the higher most data type of the variables without any intervention by the user. For example:

```
int a = 10;  
float b = 25.5f, c;  
c = a + b;
```

In case of explicit type conversion, the data gets converted to a type as specified by the programmer. For example:

```
int a = 10;
double b = 25.5;
float c = (float)(a + b);
```

Question 9

How are private members of a class different from public members?

Private members	Public members
They are accessible only within the class in which they are declared.	They are accessible both within and outside their class.
They are not inherited by derived class	They are inherited by derived class

Question 10

What are the two ways of invoking methods?

Two ways of invoking methods are:

1. **Pass by value.**
2. **Pass by reference.**

Question 11

List out the access specifiers in java?

1. **private** — private members are accessible only inside their own class.
2. **protected** — protected members are accessible inside their own class, classes within the package and subclasses.
3. **public** — public members are accessible in all the classes.

Question 12

Explain the terms operator precedence and associativity.

Operator precedence specifies the order in which the operators in an expression are evaluated when the expression has several operators.

When an expression has two operators with the same precedence, the expression is evaluated according to its associativity.

Associativity of an operator can be either right-to-left or left-to-right.

Question 13

Name the package that contains:

i) Scanner class

java.util

ii) String class

java.lang

Question 14

Name the type of error (syntax, runtime or logical error) in each case given below:

i) `Math.sqrt(25 - 50)`

Runtime Error

ii) `double x;y;z;`

Syntax Error

Question 15

Differentiate between the following

i) System.exit(0) and break

System.exit(0)	break
It is a method of System class	It is a Java language statement
It terminates the program	It terminates the loop or the switch block inside which it is placed.

ii) Null loop and an infinite loop.

Null loop	Infinite loop
It has an empty body	It may or may not have an empty body
It iterates for a finite number of iterations	It continues iterating indefinitely
Example: int i = 0; while(++i <= 10);	Example: int i = 0; while(i <= 10) System.out.println("Infinite Loop");

iii) break and continue

break	continue
It is used to unconditionally jump out of the loop	It is used to unconditionally jump to the next iteration of the loop, skipping the remaining statements of the current iteration.
It is used in switch-case and loops	It is only used in loops.

iv) Boxing-Unboxing

Boxing is the conversion of primitive data type into an object of its corresponding wrapper class. Unboxing is the opposite of Boxing, it is the conversion of wrapper class object into its corresponding primitive data type. Below program highlights the difference between the two:

```
public class Boxing {
    public static void main(String args[]) {
        int a = 100, b;
        //Boxing
        Integer aWrapped = new Integer(a);
        //Unboxing
        b = aWrapped;
        System.out.println("Boxed Value: " + aWrapped);
        System.out.println("Unboxed Value: " + b);
    }
}
```

v) Searching and Sorting

Sorting	Searching
Sorting means to arrange the elements of the array in ascending or descending order.	Searching means to search for a term or value in an array.
Bubble sort and Selection sort are examples of sorting techniques.	Linear search and Binary search are examples of search techniques.

vi) while loop and do-while loop

while	do-while
It is an entry-controlled loop.	It is an exit-controlled loop.
It is helpful in situations where numbers of iterations are not known.	It is suitable when we need to display a menu to the user.

vii) Single Dimensional Array and Double Dimensional Array

Single Dimensional Array	Double Dimensional Array
It contains single row and multiple columns.	It contains multiple rows and multiple columns.
It needs a single subscript to access its elements.	It needs two subscripts to access its elements.

viii) length and length()

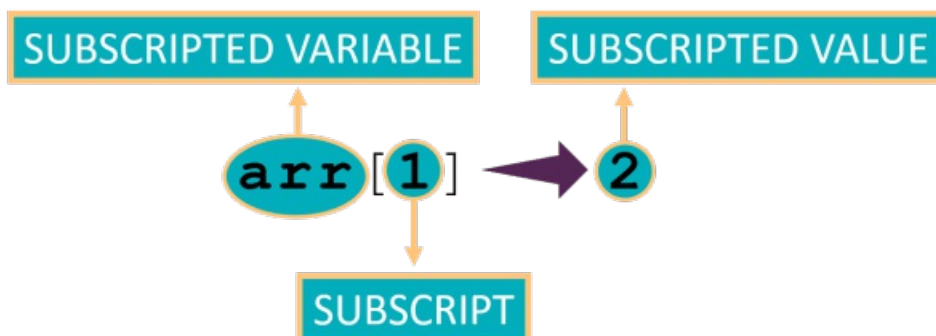
length	length()
length is an attribute i.e. a data member of array.	length() is a member method of String class.
It gives the length of an array i.e. the number of elements stored in an array.	It gives the number of characters present in a string.

ix) Linear search and Binary search

Linear Search	Binary Search
Linear search works on sorted and unsorted arrays	Binary search works on only sorted arrays (ascending or descending)
Each element of the array is checked against the target value until the element is found or end of the array is reached	Array is successively divided into 2 halves and the target element is searched either in the first half or in the second half
Linear Search is slower	Binary Search is faster

x) Subscript and subscripted variable

Subscript is the index of the element in the array whereas Subscripted variable is the name of the array when it is used with a subscript to access a single element of the array.



xi) Constructor and Method

Constructor	Method
It is a block of code that initializes a newly created object.	It is a group of statements that can be called at any point in the program using its name to perform a specific task.
It has the same name as class name.	It should have a different name than class name.
It has no return type	It needs a valid return type if it returns a value otherwise void
It is called implicitly at the time of object creation	It is called explicitly by the programmer by making a method call
If a constructor is not present, a default constructor is provided by Java	In case of a method, no default method is provided.
It is not inherited by subclasses.	It may or may not be inherited depending upon its access specifier.

xii) next() and nextLine()

next()	nextLine()
It reads the input only till the space so it can read only a single word.	It reads the input till the end of line so it can read a full sentence including spaces.
It places the cursor in the same line after reading the input.	It places the cursor in the next line after reading the input.

xiii) == and equals()

equals()	==
It is a method	It is a relational operator
It is used to check if the contents of two strings are same or not	It is used to check if two variables refer to the same object in memory
Example: <pre>String s1 = new String("hello"); String s2 = new String("hello"); boolean res = s1.equals(s2); System.out.println(res);</pre>	Example: <pre>String s1 = new String("hello"); String s2 = new String("hello"); boolean res = s1 == s2; System.out.println(res);</pre>
The output of this code snippet is true as contents of s1 and s2 are the same.	The output of this code snippet is false as s1 and s2 point to different String objects.

xiv) Syntax Error and Logical Error

Syntax Error	Logical Error
Syntax Errors occur when we violate the rules of writing the statements of the programming language.	Logical Errors occur due to our mistakes in programming logic.
Program fails to compile and execute.	Program compiles and executes but doesn't give the desired output.
Syntax Errors are caught by the compiler.	Logical errors need to be found and corrected by people working on the program.

xv) Primitive Data Types and Composite Data Types

Primitive Data Types	Composite Data Types
Primitive Data Types are Java's fundamental data types	Composite Data Types are created by using Primitive Data Types
Primitive Data Types are built-in data types defined by Java language specification	Composite Data Types are defined by the programmer
Examples of Primitive Data Types are byte, short, int, long, float, double, char, boolean	Examples of Composite Data Types are Class and Array

Evaluate the given expressions:

Question 1

```
int a=4, b=2, c=8;
```

i) $a + b * c \% a$

$$\begin{aligned}
 & \mathbf{a + b * c \% a} \\
 \Rightarrow & \mathbf{4 + 2 * 8 \% 4} \\
 \Rightarrow & \mathbf{4 + 0} \\
 \Rightarrow & \mathbf{4}
 \end{aligned}$$

ii) $a + b / c + a$

$$\begin{aligned}
 & \mathbf{a + b / c + a} \\
 \Rightarrow & \mathbf{4 + 2 / 8 + 4} \\
 \Rightarrow & \mathbf{4 + 0 + 4} \\
 \Rightarrow & \mathbf{8}
 \end{aligned}$$

iii) $a \% b + b \% c + b / a + c / b$

$$\begin{aligned}
 & \mathbf{a \% b + b \% c + b / a + c / b} \\
 \Rightarrow & \mathbf{4 \% 2 + 2 \% 8 + 2 / 4 + 8 / 2} \\
 \Rightarrow & \mathbf{0 + 2 + 0 + 4} \\
 \Rightarrow & \mathbf{6}
 \end{aligned}$$

iv) $c += a++ - ++b + a - c$

$$\begin{aligned}
 & \mathbf{c = c + (a++ - ++b + a - c)} \\
 \Rightarrow & \mathbf{c = 8 + (4 - 3 + 5 - 8)} \\
 \Rightarrow & \mathbf{c = 8 + (-2)} \\
 \Rightarrow & \mathbf{c = 8 - 2} \\
 \Rightarrow & \mathbf{c = 6}
 \end{aligned}$$

Question 2

```
int m=100, n=200, p=500;
```

i) $m + n > n - p$

$$\begin{aligned}
 & \mathbf{100 + 200 > 200 - 500} \\
 \Rightarrow & \mathbf{300 > -300} \\
 \Rightarrow & \mathbf{true}
 \end{aligned}$$

ii) $m + p \% m == n - 100$

$$\begin{aligned}
 & \mathbf{100 + 500 \% 100 == 200 - 100} \\
 \Rightarrow & \mathbf{100 + 0 == 100} \\
 \Rightarrow & \mathbf{100 == 100} \\
 \Rightarrow & \mathbf{true}
 \end{aligned}$$

iii) $!(m * 2 == n)$

$!(100 * 2 == 200)$
 $\Rightarrow !(200 == 200)$
 $\Rightarrow !(true)$
 $\Rightarrow false$

iv) $m < n \ \&\& \ p \% m == 0$

$100 < 200 \ \&\& \ 500 \% 100 == 0$
 $\Rightarrow true \ \&\& \ true$
 $\Rightarrow true$

v) $n \% m != p \% p$

$200 \% 100 != 500 \% 500$
 $\Rightarrow 0 != 0$
 $\Rightarrow false$

Determine the output of the following statements

Question 1

```
System.out.println (1+2*5%3);
```

Output

2

Explanation

$1 + 2 * 5 \% 3$
 $\Rightarrow 1 + 10 \% 3$
 $\Rightarrow 1 + 1$
 $\Rightarrow 2$

Question 2

```
System.out.println (1.5 + 2*'b');
```

Output

197.5

Explanation

$1.5 + 2 * 'b'$
 $\Rightarrow 1.5 + 2 * 98$ (ASCII Code of 'b' is 98)
 $\Rightarrow 1.5 + 196$
 $\Rightarrow 197.5$

Question 3

```
char ch = 'X';  
System.out.println(ch++ + 5);  
System.out.println(--ch);
```

Output

93
X

Explanation

```
ch++ + 5  
⇒ 88 + 5 (ch will be converted to int due to implicit casting)  
⇒ 93
```

ch is 'Y' due to ch++ in the previous statement. --ch will first decrement ch to 'X' and then print it.

Question 4

```
System.out.println(Math.pow (25,0.5));
```

Output

```
5
```

Explanation

Math.pow(25,0.5) means $\sqrt{25} \Rightarrow 5$

Question 5

```
System.out.println (Math.max('C','G'));
```

Output

```
71
```

Explanation

As return type of Math.max is int so 'G' will be converted to its ASCII code which is 71.

Question 6

```
System.out.println ("ABCD" + 10 + 10.5);
```

Output

```
ABCD1010.5
```

Explanation

As "ABCD" is String so implicit type conversion will convert 10 and 10.5 also to String and add it to the end of "ABCD".

Question 7

```
int m=2, n=10;  
System.out.println(m * 2 == n);  
System.out.println(!(m<n && n>m));
```

Output

```
false  
false
```

Explanation

$m * 2 \Rightarrow 2 * 2 \Rightarrow 4$. So $m * 2 == n$ is false.

$m < n \ \&\& \ n > m$ is true so $!(m < n \ \&\& \ n > m)$ becomes false.

Question 8

```
System.out.println("LAPTOP".charAt(0) + "LENOVO".charAt(1));
```

Output

```
145
```

Explanation

```
"LAPTOP".charAt(0) + "LENOVO".charAt(1) ⇒ 'L' + 'E'.
```

As we are trying to add two char type values, both will be converted to their respective ASCII codes and the output will be the sum of their ASCII codes as an int value.

'L' + 'E' ⇒ 76 + 69 ⇒ 145

Question 9

```
System.out.println(Character.toUpperCase("CLOcK".charAt(3)));
```

Output

```
c
```

Explanation

```
"CLOcK".charAt(3) gives 'c'.
```

```
Character.toUpperCase('c') returns 'C'.
```

Question 10

```
System.out.println(Character.isLetter('j'));
```

Output

```
true
```

Explanation

As 'j' is a letter so `Character.isLetter('j')` returns true.

Analyze how many times the following loops gets executed and write the output

Question 1

```
int x=100;
while (x!=0)
{
    System.out.println(x);
    if (x % 6 == 0)
        break;
    x--;
}
```

Output

```
100
99
98
97
96
```

Explanation

Loop executes 5 times.

When x becomes 96, x % 6 is 0 so break statement is executed and loop exits.

Question 2

```
int m=2, n=5;
while (m <= n)
{
    System.out.println(m + " " + n);
    m++; n++;
}
```

Output

```
2 5
3 6
. .
. .
. .
```

Explanation

Infinite loop as m and n both are incremented by 1 so m <= n is always true.

Question 3

```
int i = 0;
while (++i <= 10)
{
    if (i % 2 == 0)
        continue;
    System.out.println(i);
}
```

Output

```
1
3
5
7
9
```

Explanation

Loop executes 10 times.

Due to the if check, for even numbers continue statement gets executed because of which println is skipped.

Question 4

```
int i;
for (i = 50; i >= 10; i -= 10)
System.out.println(i);
System.out.println(++i*10);
```

Output

```
50
40
30
20
10
10
```

Explanation

Loop executes 5 times.

As there are no curly brackets in the for loop so only `System.out.println(i);` is in the loop body. It gets executed 5 times. As `i` is decremented by 10 in each iteration, when the loop exits, the value of `i` is 0. So, $++i*10 \Rightarrow 1 * 10 \Rightarrow 10$.

Question 5

```
for (int i=3; i>=1; i--)
{
    for (int j=i; j >= 1; j--)
        System.out.println(j);
}
```

Output

```
321
21
1
```

Explanation

Outer loop executes 3 times, inner loop executes 6 times.

i	j	Output	Remarks
3	3	3	1 st Iteration of outer loop
3	2	32	
3	1	321	
2	2	321 2	2 nd Iteration of outer loop
2	1	321 21	
1	1	321 21 1	3 rd Iteration of outer loop

Write the Output
Question 1

```
String s1="BASIC", s2="BASE";
System.out.println(s1.length() + s2.length());
System.out.println(s1.charAt(0) == s2.charAt(0));
System.out.println(s2.charAt(s1.length() - 3));
System.out.println(s1.startsWith(s2.substring(0,2)));
System.out.println(s1.concat(""+s2));
System.out.println(s1.substring(0,1).concat(s2.substring(1)));
System.out.println(s1.compareTo(s2));
System.out.println(s1.toLowerCase().charAt(3));
System.out.println(s2.substring(2, s2.length()));
System.out.println(s1.replace(s1.charAt(s1.length() - 1), 'L'));
```

Output

```
9
true
5
true
BASICBASE
BASE
4
i
SE
BASIL
```

Question 2

```
int arr[] = {12, 10, 5, 8, 7};
System.out.println(arr.length + arr[arr.length - 1]);
System.out.println(arr[0]++ + --arr[2] * arr[4]);
for (int i = 0; i < arr.length; i++)
System.out.print(arr[i] + " ");
System.out.println("\n is the output");
```

Output

```
12
40
13 10 4 8 7
is the output
```

Question 3

```
char x[] = {65, 66, 67, 68, 32}; //Note: 32 is the ASCII of space char
System.out.println(x[0] + " " + x[3]);
System.out.println(++x[1]);
System.out.println(x[1]);
for (int i = 0; i < x.length; i++)
System.out.println(x[i]);
```

Output

```
A D
C
C
A
C
C
D
```

Question 4

```
String st[] = {"MARS", "EARTH", "MOON", "MERCURY"};
System.out.println(st.length);
System.out.println(st[0].length() > st[2].length());
System.out.println(st[1].indexOf('S'));
System.out.println(st[3].charAt(0) == st[2].charAt(0));
System.out.println(st[0].charAt(0) + "" + st[1].charAt(0));
System.out.println(st[1].charAt(1) + st[2].charAt(0));
System.out.println(st[2].length() + st[3].length());
```

Output

```
4
false
-1
true
ME
142
11
```

Question 5

```
double x[]={1, 2, 3, 4.56}; int a=2;
System.out.println(x[1]);
System.out.println(Double.toString(x[2] + x[1]));
System.out.println(x[0] + x[1] * x[2]);
System.out.println(++x[++a]);
```

Output

```
2.0
5.0
7.0
5.56
```

Question 6

```
String s="Examination";
int a=20, b=19;
System.out.println(s+a+b);
System.out.println(a+b+s);
System.out.println('A'+B+" is the Output");
System.out.println("Output is " + 'A'+B');
System.out.println("Output is " + ('A'+B'));
System.out.println("Output is " + a*b%2);
System.out.println("Output is "+ a'+5);
System.out.println('a' + 5 + " =Output");
```


Output

```
Examination2019
39Examination
131 is the Output
Output is AB
Output is 131
Output is 0
Output is a5
102=Output
```

Rewrite as directed

Question 1

Using single if statement:

```
if (code == 'G')
System.out.println("Green");
if (code == 'g')
System.out.println("Green");
```

Answer

```
if (code == 'G' || code == 'g')
    System.out.println("Green");
```

Question 2

Using while loop:

```
for (int x = 1, y = 2; x <= 5 && y <= 6; x++, y++)
System.out.println(x + "\n" + y);
```

Answer

```
int x = 1, y = 2;
while (x <= 5 && y <= 6) {
    System.out.println(x + "\n" + y);
    x++;
    y++;
}
```

Question 3

Using do...while loop:

```
for (int a = 125; a > 0; a = a / 10)
{
    System.out.println(a % 10);
}
```

Answer

```
int a = 125;
do {
    System.out.println(a % 10);
    a = a / 10;
} while (a > 0);
```

Question 4

Using ternary operator:

```
if (a % 2 == 0)
System.out.println("EVEN");
else
System.out.println("ODD");
```

Answer

```
System.out.println(a % 2 == 0 ? "EVEN" : "ODD");
```

Question 5

Using nested if statement:

```
if (a >= 10 && a <= 20)
System.out.println("Number in range");
else
System.out.println("Number not in range");
```

Answer

```
if (a >= 10) {
    if (a <= 20) {
        System.out.println("Number in range");
    }
}
else {
    System.out.println("Number not in range");
}
```