**Question 1**

What is the result of attempting to compile and run the following code?

**public** **class** Outer {

**private** String x = "Outer variable";

**void** doStuff() {

String z = "local variable";

**class** Inner {

**public** **void** seeOuter() {

System.*out*.println("Outer x is " + x);

System.*out*.println("Local variable z is " + z);

}

}

}

}

**Answer:** Compilation Error

Cannot refer to a non-final variable z inside an inner class defined in a different method.

**Question 2**

What is the Synchronized resizable-array implementation of the List interface is?

**Answer:** Vector

Vector implements List, RandomAccess - Synchronized resizable-array implementation of the List interface with additional "legacy methods."

**Question 3**

What is the result of attempting to compile and run the following code?

**public** **class** Test {

**static** {

**int** a = 5;

}

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

System.*out*.println(a);

}

}

**Anwser:** Compilation Error

A variable declared in a static initialiser is not accessible outside its enclosing block.

**Question 4**

What is the result of attempting to compile and run the following code?

**static** **public** **class** Test {

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

**char** c = 'a';

**switch** (c) {

**case** 65:

System.*out*.println("one");

**break**;

**case** 'a':

System.*out*.println("two");

**break**;

**case** 3:

System.*out*.println("three");

}

}

}

**Answer:** Compile error - Illegal modifier for the class Test; only public, abstract & final are permitted.

**Question 5**

What is the result of attempting to compile and run the following code?

**public** **class** Test {

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

**int** x = 3;

**int** y = 1;

**if** (x = y)

System.*out*.println("Not equal");

**else**

System.*out*.println("Equal");

}

}

**Answer:** An error at " if (x = y)" causes compilation to fall.

**Question 6**

What is a transient variable?

**Answer:** A transient variable is a variable that may not be serialized.

**Question 7**

How can a subclass call a method or a constructor defined in a superclass?

**Answer:** Using super()

**Question 8**

There are two classes: A and B. The class B need to inform class A when some important event has happened. What Java technique would you use to implement it?

**Answer:** If these classes are threads, I'd consider notify() or notifyAll(). For regular classes you can use the Observer interface.

**Question 9**

What would you use to compare two String variables - the operator == or the method equals()?

**Answer:** I'd use the method equals() to compare the values of the Strings and the == to check if two variables point at the same instance of a String object.

**Question 10**

How would you make a copy of an entire Java object with its state?

**Answer:** Have this class implement Cloneable interface and call its method clone().

**Question 11**

What can you do in your Servlet/JSP code to tell browser not to cache the pages?

**Answer:**

You will need to set the appropriate HTTP header attributes to prevent the dynamic content output by the JSP page from being cached by the browser. Just execute the following scriptlet at the beginning of your JSP pages to prevent them from being cached at the browser. You need both the statements to take care of some of the older browser versions.

<%  
response.setHeader("Cache-Control","no-store"); //HTTP 1.1  
response.setHeader("Pragma\","no-cache"); //HTTP 1.0  
response.setDateHeader ("Expires", 0); //prevents caching at the proxy server  
%>

**Question 12**

What is the difference between request parameters and request attributes?

**Answer:**

Attributes are objects, and can be placed in the request, session, or context objects. Because they can be any object, not just a String, they are much more flexible. You can also set attributes programmatically and retrieve them later.

**Question 13**

What is the use of ActionForm class. Describe its Life Cycle?

**Answer:**

ActionForm class is used to capture user-input data from an HTML form and transfer it to the Action Class. ActionForm plays the role of Transport Vehicle between the presentation Tier & Business Tier.

Life Cycle : 1. Request received by Controller; 2. Create or recycle ActionForm; 3. Call reset(); 4. store ActionForm in proper scope; 5. populate ActionForm from Request; 6. Validate the ActionForm; 7. If errors found then forward back to input attribute page(configured in Action mapping in struts-config.xml) with ActionForm in scope. If no errors found then call execute() with the ActionForm.

**Question 14**

What is the difference between Action and DispatchAction classes?

**Answer:** Grouping related actions into one class is possible using DispatchAction class. Action class is used to perform single functionality.

**Question 15**

What is the difference between web.xml and sturts-config.xml?

**Answer:** web.xml is utilized to make connections between web application and the web container, where as the struts-config.xml is utilized to establish the connections between the controller and the view of MVC architecture.

At the time of starting the container, the web.xml will be read by the container, whereas the struts-config.xml will be read by the ActionServlet’s init() method.

**Question 16**

Explain Quick Sort with code snippet & its complexity?

**Answer:**

Θ(n log(n))

It works recursively by a divide-and-conquer strategy.

void quicksort (int[] a, int lo, int hi)

{

// lo is the lower index, hi is the upper index

// of the region of array a that is to be sorted

int i=lo, j=hi, h;

// comparison element x

int x=a[(lo+hi)/2];

// partition

do

{

while (a[i]<x) i++;

while (a[j]>x) j--;

if (i<=j)

{

h=a[i]; a[i]=a[j]; a[j]=h;

i++; j--;

}

} while (i<=j);

// recursion

if (lo<j) quicksort(a, lo, j);

if (i<hi) quicksort(a, i, hi);

}

**Question 17**

Write an algorithm to find whether the given string is a palindrome?

Public class palindrome  
{  
Boolean isPalindrome(String testString){  
    return(testString.Equals(reverse(testString)));  
}  
  
public static String reverse(String s){  
    String and;  
      
    if (s.Length() <= 1)  
        return s;  
      
    else{     
        char lastC = s.CharAt(s.Length()-1);          
        String stringLeft = s.Substring(0, s.Length() -1);  
        return and = lastC + reverse(stringLeft);     
        }  
    }  
}

**Question 18**

Write an iterative function to find the position of the element in an array using binary search?

  public static int binarySearch( Comparable [ ] a, Comparable x )  
    {  
        int low = 0;  
        int high = a.length - 1;  
        int mid;  
  
        while( low <= high )  
        {  
            mid = ( low + high ) / 2;  
  
            if( a[ mid ].compareTo( x ) < 0 )  
                low = mid + 1;  
            else if( a[ mid ].compareTo( x ) > 0 )  
                high = mid - 1;  
            else  
                return mid;  
        }  
  
        return NOT\_FOUND;

}

**Question 19**

Write a recursive algorithm to get the Fibonacci series?

**public** **class** Fibonacci **{**

**public** **static** **int** fib**(int** n**)** **{**

**if** **(**n < 2**)** **{**

**return** n;

**}**

**else** **{**

**return** fib**(**n-1**)**+fib**(**n-2**)**;

**}**

**}**

**}**

**Question 20**

Write an iterative function to find the position of the element in an array using binary search?

public static int **binarySearch( Comparable [ ] a, Comparable x )  
    {**int **low = 0;**int **high = a.length - 1;**int **mid;**while**( low <= high )  
        {  
            mid = ( low + high ) / 2;**if**( a[ mid ].compareTo( x ) < 0 )  
                low = mid + 1;**else if**( a[ mid ].compareTo( x ) > 0 )  
                high = mid - 1;**elsereturn **mid;  
        }**return **NOT\_FOUND**;

    }