**Time allowed: 90 Minutes Max. Marks: 40**

**General Instructions:**

* **Follow the instructions given in each section.**
* **Make sure that you attempt the questions in order.**

**SECTION-A (10\*1 mark=10 marks)**

***(All questions are compulsory)***

1)What is the syntax of abstract class in java?

a) abstract A{}

b) abstract class A

**c) abstract class A{}**

d) abstract class A[]

2) Which of these packages contains abstract keyword?

**a) java.lang**

b) java.util

c) java.io

d) java.system

3) For which purpose packages are used in Java?

a) Categorizes data

**b) Organizing java classes into namespaces**

c) For faster compilation

d) None of these

4) The first statement in java source file

a) import statement

**b) package statement**

c) main statement

d) try{} and catch{}

5) What will be the output of below statements?

String s = "Java"+1+2+"Quiz"+""+(3+4);

System.out.println(s);

a) Java3Quiz7

**b) Java12Quiz7**

c) Java12Quiz34

d) Java3Quiz34

6) How many String objects created in below statements?

String s = "abc"; // statement 1

String s1 = new String("abcd"); // statement 2

a) 1

b) 2

**c) 3**

d) 4

7) What will be the output of below statements?

String s1 = "abc";

String s2 = new String("abc");

System.out.print(s1==s2);

System.out.println(s1==s2.intern());

**a) falsetrue**

b) falsefalse

c) truetrue

d) truefalse

8) Select all the reasons that make String perfect candidate for Map key?

a) String is immutable

b) String is final

c) String properly implements hashCode() and equals() method

**d) All of the above**

9) Which of the following are incorrect form of StringBuffer class constructor?

a) StringBuffer()

b) StringBuffer(int size)

c) StringBuffer(String str)

**d) StringBuffer(int size , String str)**

10) What happen in case of multiple catch blocks?

a) Either super or subclass can be caught first.

b) The superclass exception must be caught first.

**c) The superclass exception cannot caught first.**

d) None of these

**SECTION-B (5\*2 mark=10 marks)**

***(All questions are compulsory)***

11) What is the output of the following program code?

public class Test{

public static void main(String args[]){

try{

int i;

return;

}

catch(Exception e){

System.out.print("inCatchBlock");

}

finally{

System.out.println("inFinallyBlock");

}

}

}

a) inCatchBlock

b) inCatchBlock inFinallyBlock

**c) inFinallyBlock**

d) The program will return without printing anything

12) What is the result of the following program?

public static synchronized void main(String[] args) throws

InterruptedException {

Thread f = new Thread();

f.start();

System.out.print("A");

f.wait(1000);

System.out.print("B");

}

a) It prints A and B with a 1000 seconds delay between them

b) It only prints A and exits

c) It only prints B and exits

**d) A will be printed, and then an exception is thrown.**

13) Determine output:

public class Test

{

public static void main(String args[])

{

String s1 = "SITHA";

String s2 = "RAMA";

System.out.println(s1.charAt(0) > s2.charAt(0));

}

}

a) 0

**b) True**

c) False

d) Compilation error

14) What is the output of this program?

Note : Output.class file is not in directory pkg.

package pkg;

class output

{

public static void main(String args[])

{

StringBuffer s1 = new StringBuffer("Hello World");

s1.insert(6 , "Good ");

System.out.println(s1);

}

}

a) HelloGoodWorld

b) HellGoodoWorld

c) Compilation error

**d) Runtime error**

15) What will be the output of the following Java code?

class A

{

public int i;

private int j;

}

class B extends A

{

void display()

{

super.j = super.i + 1;

System.out.println(super.i + " " + super.j);

}

}

class inheritance

{

public static void main(String args[])

{

B obj = new B();

obj.i=1;

obj.j=2;

obj.display();

}

}

a) 2 2

b) 3 3

c) Runtime Error

**d) Compilation Error**

**SECTION-C(Coding Question) (2x5 marks=5 marks)**

Q16) Given two strings A and B. Find the characters that are not common in the two strings.

Note :- Return the string in sorted order.

**Input :**

A = characters

B = alphabets

**Output:** bclpr

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test Case 1** | **Test Case 2** | **Test Case 3** |
| **Input** | A= Science  B= English | A= Pandharpur  B= Akkalkot | A= Pizza  B= pasta |
| **Output** | ESceghls | APdhklnoprtu | Pipstz |

Solution :

**import java.io.\*;**

**import java.util.\*;**

**public class MyClass**

**{**

**public static void main(String args[])throws IOException**

**{**

**String A = "India";**

**String B = "America";**

**Solution ob = new Solution();**

**System.out.println(ob.UncommonChars(A, B));**

**}**

**}**

**class Solution**

**{**

**String UncommonChars(String A, String B)**

**{**

**String uncommon="";**

**int i=0;**

**while(i<A.length()){**

**if(!B.contains(Character.toString(A.charAt(i)))){**

**if(!uncommon.contains(Character.toString(A.charAt(i)))){**

**uncommon+=A.charAt(i);**

**}**

**}**

**i++;**

**}**

**i=0;**

**while(i<B.length()){**

**if(!A.contains(Character.toString(B.charAt(i)))){**

**if(!uncommon.contains(Character.toString(B.charAt(i)))){**

**uncommon+=B.charAt(i);**

**}**

**}**

**i++;**

**}**

**if(uncommon==""){**

**return "-1";**

**}**

**char a[] = uncommon.toCharArray();**

**Arrays.sort(a);**

**String str = new String(a);**

**return str;**

**}**

**}**

Q17) Given a sentence in the form of a string in uppercase, convert it into its equivalent mobile numeric keypad sequence. Please note there might be spaces in between the words in a sentence and we can print spaces by pressing 0.

**Input**: S = "MUG"

**Output**: 6884

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test Case 1** | **Test Case 2** | **Test Case 3** |
| **Input** | S = "HEY U" | S = "PIZZA" | S = "AMAZON" |
| **Output** | 4433999088 | 7444999999992 | 262999966666 |

Solution :

**import java.io.\*;**

**import java.util.\*;**

**public class MyClass{**

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

**String S ="MUG";**

**Solution obj = new Solution();**

**System.out.println(obj.printSequence(S));**

**}**

**}**

**class Solution**

**{**

**String printSequence(String S)**

**{**

**String str[]={"2","22","222","3","33","333","4","44","444","5","55","555","6","66","666",**

**"7","77","777","7777","8","88","888","9","99","999","9999"};**

**String result="";**

**for(int i=0;i<S.length();i++){**

**if(S.charAt(i)!=' ')**

**result+=str[S.charAt(i)-'A'];**

**else**

**result=result+'0';**

**}**

**return result;**

**}**

**}**

**SECTION-D (Coding Question)(1x10 mark=10 mark)**

Q18) Given two strings, one is a text string and the other is a pattern string. The task is to print the indexes of all the occurrences of the pattern string in the text string. For printing, Starting Index of a string should be taken as 1.

**Input**: s= extraexclusive

patt=ex

**Output**: 1 6

Solution :

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test Case 1** | **Test Case 2** | **Test Case 3** |
| **Input** | s = “chotu-motu“  patt = “tu” | s = “evergreen“  patt = “e” | s = “moonbloom“  patt = “oo” |
| **Output** | 4 9 | 1 3 7 8 | 2 7 |

Solution**:**

**import java.io.\*;**

**import java.util.\*;**

**public class MyClass**

**{**

**public static void main(String args[])throws IOException**

**{**

**String s="extraexclusive";**

**String patt="ex";**

**Solution ob = new Solution();**

**ArrayList<Integer> res = ob.search(patt, s);**

**if(res.size()==0)**

**System.out.print("-1 ");**

**else {**

**for(int i = 0;i<res.size();i++)**

**System.out.print(res.get(i) + " ");**

**}**

**}**

**}**

**class Solution**

**{**

**ArrayList<Integer> search(String pat, String S)**

**{**

**ArrayList<Integer> arr=new ArrayList<>();**

**for(int i=0; i<=S.length()-pat.length(); i++){**

**if(S.substring(i,i+pat.length()).equals(pat))**

**arr.add(i+1);**

**}**

**return arr;**

**}**

**}**