Java Assignment 1

Logical Coding Question:

1)

```
   *
  * *
 *   *
*     *
*********
```

2)

```
*********
*     *
 *   *
  * *
   *
```

3)

```
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
```

4)

```
          1
        1   1
      1   2   1
    1   3   3   1
  1   4   6   4   1
```

5)

```
   1
  212
 32123
4321234
 32123
  212
   1
```

6)

```
1
2 1
3 2 1
4 3 2 1
5 4 3 2 1
```

7)

```
10101
01010
10101
01010
10101
```

8)

```
1
10
101
1010
10101
```

9)

```
1 2 3 4 5
 2 3 4 5
  3 4 5
   4 5
    5
   4 5
  3 4 5
 2 3 4 5
1 2 3 4 5
```

10)

```
A
B B
C C C
D D D D
E E E E E
F F F F F F
```

11)

```
A B C D E F
A B C D E
A B C D
A B C
A B
A
A
A B
A B C
A B C D
A B C D E
A B C D E F
```

12)

```
     A
    A B
   A B C
  A B C D
 A B C D E
A B C D E F
```

13)

```
     A
    B B
   C   C
  D     D
 E       E
F         F
 E       E
  D     D
   C   C
    B B
     A
```

15)

```
1 2 3 4 5 6
 2 3 4 5 6
  3 4 5 6
   4 5 6
    5 6
     6
```

16) Printing Christmas Tree Using Pyramid

```
                *
               * *
              * * *
             * * * *
            * * * * *
             * * *
             * * * *
            * * * * *
           * * * * * *
          * * * * * * *
           * * * * *
           * * * * * *
          * * * * * * *
         * * * * * * * *
        * * * * * * * * *
         * * * * * * *
         * * * * * * * *
        * * * * * * * * *
       * * * * * * * * * *
      * * * * * * * * * * *
       * * * * * * * * *
       * * * * * * * * * *
      * * * * * * * * * * *
     * * * * * * * * * * * *
    * * * * * * * * * * * *
*       * * * * * * * * * * *
* *           * * * *
            * * * *
            * * * *
            * * * *
17)

*

* *

* * *

* * * *

* * * * *

* * * *
```

```
* * *

* *

*

18)

7
7 6
7 6 5
7 6 5 4
7 6 5 4 3
7 6 5 4 3 2
7 6 5 4 3 2 1

19)

1
1 2 1
1 2 3 2 1
1 2 3 4 3 2 1
1 2 3 4 5 4 3 2 1
1 2 3 4 5 6 5 4 3 2 1
1 2 3 4 5 6 7 6 5 4 3 2 1

20)

1234567
 234567
  34567
   4567
    567
     67
      7
     67
    567
   4567
  34567
 234567
1234567

21)
```

```
1

10

101

1010

10101

101010

1010101

22)

1111111
1111122
1111333
1114444
1155555
1666666
7777777

23)
1010101
0101010
1010101
0101010
1010101
0101010
1010101

24)

1
2 6
3 7 10
4 8 11 13
5 9 12 14 15

25)

1 2 3 4 5 6 7
2 3 4 5 6 7 1
3 4 5 6 7 1 2
4 5 6 7 1 2 3
5 6 7 1 2 3 4
6 7 1 2 3 4 5
7 1 2 3 4 5 6
```

Class, Objects, Variables, Methods & Constructors Coding Questions

1. Java Program to Illustrate Use of All Features of Abstract Class
2. Java Program to Illustrates Use of Abstract Class and Method
3. Java Program to Illustrates Use of Instance Inner Class
4. Java Program to Illustrates Use of Static Inner Class
5. Java Program to Illustrates Use of Referencing the Object from Inner Class
6. Java Program to Create Outer Class Bank Account and the Inner Class Interest in it
7. Java Program to Implement Shape Interface using Circle and Rectangle Class
8. Java Program to Count Number of Objects Created for Class
9. Java Program to Implement the Passing and Returning Objects
10. Java Program to Swap Objects using Swap() Method
11. Java Program to Illustrate Use of Methods in a Class
12. Java Program to Create a Method without Parameters and with Return Type
13. Java Program to Create a Method without Parameters and Return Type
14. Java Program to Create a Method with 2 Parameters and without Return Type
15. Java Program to Illustrate the Use of HashCode() Method
16. Java Program to Illustrate Use of Final Keyword
17. Java Program to Illustrate Use of Constructor
18. Java Program to Illustrates Use of Chaining Constructor
19. Java Program to Create an Object for Class and Assign Value in the Object using Constructor
20. String Constructor Program in Java
21. Java Program to Allocate and Initialize Super Class Members using Constructor
22. Java Program to Check the Accessibility of Static and Non-Static Variable by a Static Method
23. Java Program to Demonstrate Usage of an Instance Variable in the Test Class
24. Java Program to Demonstrate Usage of a Static Variable in the Test Class
25. Java Program to Check Whether Which One is Executed First, Static Block or the Static Method
26. Java Program to Calculate Sum of Two Byte Values using Type Casting

Arrays Coding Questions:

1. Write a program to print elements of Array ?
2. Write a Java program to check the equality of two arrays?
3. Write a Java program to find all pairs of elements in an integer array whose sum is equal to a given number?
4. Write a program to reverse an Array in java .
5. Find out smallest and largest number in a given Array?
6. Print the third-largest number in an array without sorting it
   Input: [24,54,31,16,82,45,67]

Output: 54 (82 and 67 are the largest and second-largest)

7. Write a program to merge two arrays of integers by reading one number at a time from each array until one of the array is exhausted, and then concatenating the remaining numbers.
   Input: [23,60,94,3,102] and [42,16,74]
   Output: [23,42,60,16,94,74,3,102]
8. Write a program which takes an array of integers and prints the running average of 3 consecutive integers.

   In case the array has fewer than 3 integers, there should be no output.

   Input: [5,14,35,89,140]

   Output: [18, 46, 88]

   (Explanation: 18=(5+14+35/3, 46=(14+35+89)/3, ...)

9. Write a program which generates the series 1,4,27,16,125,36
10. Given an array of integers, print whether the numbers are in ascending order or in descending order or in random order without sorting

    Input: [5,14,35,90,139] Output: Ascending

    Input: [88,67,35,14,-12] Output: Descending

    Input: [65,14,129,34,7] Output: Random

11. How to convert a byte array to String?
12. How to rotate an array left and right by a given number K?
13. Write a program to sort an Array in Java ?
14. Write a program to check whether two given Arrays are equal, given both contains same data type and same length ?
15. How to find the missing number in a given Array from number 1 to 100 ?
16. Given two arrays, find the intersection between them?
17. Find the missing number in an Array between 1 to 100. Given only one number is missing
18. How to find duplicate elements in a given Array
19. Write a program to sum all the values of a given Array in java?
20. How do you separate zeros and non-zeros in a given Array in java?
21. How to convert Array to ArrayList in java ?
22. How to convert Array to TreeSet in java ?
23. How to convert ArrayList to String Array in java ?
24. Write a program to find second largest element in a given Array in java?

String Coding Questions:

1. How to Print duplicate characters from String?
2. For example, if String is "Java" then the program should print "a"
3. How to check if two Strings are anagrams of each other?
4. How to program to print the first non repeated character from String?
5. How to check if a String contains only digits?
6. How to find duplicate characters in a String?
7. You need to write a program to print all duplicate character and their count in Java. For example, if given String is "Programming" then your program should print

   g : 2

   r : 2

   m : 2

8. How to count the occurrence of a given character in String?
9. How to convert numeric String to an int?
10. For example, if you pass "67263" to the program then it should return 67263.
11. How to replace each given character to other e.g. blank with %20?
12. For example, if the input is "Java is Great" and asked to replace space with %20 then it should be "Java%20is%20Great".
13. How to find all permutations of String?
14. How to reverse words in a sentence without using a library method?
15. How to remove duplicate characters from String?
16. For example, if the input is ‘bananas’ the output will be ‘bans’.
17. How to check if a String is a valid shuffle of two String?
18. For example, given first = "abc" and second = "def", third = "dabecf" is a valid shuffle since it preserves the character ordering of the two strings.
19. How to return the highest occurred character in a String?
    For example if input is "aaaaaaaaaaaaaaaaabbbbcddddeeeeee" it should return "a".
20. Write a program to remove a given character from String?
21. You need to write a Java method that will accept a String and a character to be removed and return a String, which doesn't has that character e.g remove(String word, char ch).
22. Write a program to find the longest palindrome in a string?
23. How to sort String on their length in Java?
24. Write a Program to sort String on their length in Java? Your method should accept an array of String and return a sorted array based upon the length of String. Don't forget to write unit tests for your solution.

25. This question is asked by Facebook. Given a string, return whether or not it forms a palindrome ignoring case and non-alphabetical characters.

Special Coding questions:

1. Note: a palindrome is a sequence of characters that reads the same forwards and backwards.

   Ex: Given the following strings...

   "A man, a plan, a canal: Panama.", return true

2. This question is asked by Google. Given a string, return whether or not it uses capitalization correctly. A string correctly uses capitalization if all letters are capitalized, no letters are capitalized, or only the first letter is capitalized.

   Ex: Given the following strings...

   "USA", return true

   "Calvin", return true

   "compUter", return false

   "coding", return true

3. This question is asked by Amazon. Given a string representing the sequence of moves a robot vacuum makes, return whether or not it will return to its original position. The string will only contain L, R, U, and D characters, representing left, right, up, and down respectively.

   Ex: Given the following strings...

   "LR", return true

   "URURD", return false

   "RUULLDRD", return true

4. Given two binary strings (strings containing only 1s and 0s) return their sum (also as a binary string).
5. Note: neither binary string will contain leading 0s unless the string itself is 0

   Ex: Given the following binary strings...

   "100" + "1", return "101"

"11" + "1", return "100"

"1" + "0", return "1"

6. Given an array of strings, return the longest common prefix that is shared amongst all strings.
7. Note: you may assume all strings only contain lowercase alphabetical characters.

   Ex: Given the following arrays...

   ["colorado", "color", "cold"], return "col"

   ["a", "b", "c"], return ""

   ["spot", "spotty", "spotted"], return "spot"

8. Given a string and the ability to delete at most one character, return whether or not it can form a palindrome.
9. Note: a palindrome is a sequence of characters that reads the same forwards and backwards.

   Ex: Given the following strings...

   "abcba", return true

   "foobof", return true (remove the first 'o', the second 'o', or 'b')

   "abccab", return false

10. Given a string representing your stones and another string representing a list of jewels, return the number of stones that you have that are also jewels.

    Ex: Given the following jewels and stones...

    jewels = "abc", stones = "ac", return 2

    jewels = "Af", stones = "AaaddfFf", return 3

    jewels = "AYOPD", stones = "ayopd", return 0

11. Given two strings, s and t, merge the two strings together alternating characters starting with s.

Note: If one string is longer than the other, append the remaining characters of the longer string at the end of the merged string.

s = "abc", t = "def", return "adbecf".

12. Given a string, s, return the total number of substring within s that contain the same character.

Note: You may assume that s only contains lowercase alphabetical characters.

Ex: Given the following string s…

s = "aabca", return 6 ("a" appears 3 times, "aa" appears 1 time, "b" appears 1 time, and "c" appears 1 time).

13. Given an encoded string, s, return its decoded representation. The string s will be encoded as follows N[letters], meaning that the letters should be repeated N times in the decoded representation.

    Note: You may assume s always encoded correctly (i.e. correct formatting of brackets, only positive values outside the brackets, and always lowercase alphabetical characters inside the brackets).

    Ex: Given the following string s…

    s = "3[a]2[b]1[c]", return "aaabbc" ("a" is repeated 3 times, "b" is repeated 2 times, and "c" is repeated 1 time).

14. You are given a list of strings, times, where each string represent a timestamp of a twenty-four hour clock (i.e. hours and minutes - “HH:MM”). Return the minimum difference, in minutes, between any two of the timestamps in the list.

    Ex: Given the following times…

    times = ["01:00", "01:10"], return 10 (i.e. there are 10 minutes between the two times).

    Ex: Given the following times…

    times = ["00:00", "12:23", "05:50", "23:12"], return 48.

15. Given a string, s, return the length of the longest substring that contains every vowel occurring an even number of times.

    Note: You may assume s only contains lowercase alphabetical characters and the vowels you must account for are a, e, i, o, and u.

    Ex: Given the following string s…

s = "aeiouaeioua", return 10 (the last 'a' cannot count).

Ex: Given the following string s…

s = "bbb", return 3 (all vowels occur an even number of times, i.e. zero times each).

16. You are given a list of words and asked to find the longest chain. Two words (or more) form a chain if a single letter can be added anywhere in a word, s, to form a word, t (where s and t are both words within the list of words you're given). Return the length of the longest chain you can form.

    Ex: Given the following words…

    words = ["a", "ab", "abc"], return 3 ("a" can be transformed to "ab" by adding a "b" and "ab" can be transformed by adding a "c" giving a chain length of 3).

    Ex: Given the following words…

    words = ["a", "abc"], return 1 (both "a" or "abc" form their own chains of size one, they cannot be added together).

17. You are given two string arrays, queries and words. For any string, s, f(s) is equal to the number of times the smallest lexicographical characters occurs in s. For each query, queries[i] count the total number of words where f(queries[i]) < f(word) and return the answer as an array.

    Note: Both queries and words will only contain lowercase alphabetical characters and contain at most ten strings each.

    Ex: Given the following queries and words…

    queries = ["abc"], words = ["def"], return 0 ('a' and 'd' both occur once so f("abc") and f("def") are equal).

    Ex: Given the following queries and words…

    queries = ["abc"], words = ["ddef", "xxyz"], return 2 ('a' appears once and 'd' and 'x' appear twice so f("abc") is less than both f("ddef") and f("xxyz")).

Collection coding questions:

1. Write a Java program to create a new array list, add some colors (string) and print out the collection.

2. Write a Java program to insert an element into the array list at the first position.

3. Write a Java program to retrieve an element (at a specified index) from a given array list.

4. Write a Java program to sort a given array list.

5. Write a Java program to reverse elements in a array list.

6. Write a Java program of swap two elements in an array list.

7. Write a Java program to print all the elements of a ArrayList using the position of the elements.

8. Write a Java program to append the specified element to the end of a linked list.

9. Write a Java program to insert the specified element at the specified position in the linked list.

10. Write a Java program to insert elements into the linked list at the first and last position.

11. Write a Java program to display the elements and their positions in a linked list.

12. Write a Java program to check if a particular element exists in a linked list.

13. Write a Java program to compare two linked lists.

14. Write a Java program to replace an element in a linked list.

15. Write a Java program to iterate through all elements in a hash list.

16. Write a Java program to empty an hash set.

17. Write a Java program to convert a hash set to an array.

18. Write a Java program to compare two sets and retain elements which are same on both sets.

19. Write a Java program to create a new tree set, add some colors (string) and print out the tree set.

20. Write a Java program to find the numbers less than 7 in a tree set.

21. Write a Java program to remove all the elements from a priority queue.

22. Write a Java program to count the number of key-value (size) mappings in a map

23. Write a Java program to convert a priority queue to an array containing all of the elements of the queue

24. Write a Java program to check whether a map contains key-value mappings (empty) or not

25. Write a Java program to get the value of a specified key in a map