16-01-24 Strings

1) String constructor : BMSCE
abcd
String constructor using object : abcd
String constructor with index of chars : bcd
String constructor using ascii values of characters : ABCDEF
String constructor using ascii : BCD

2) String length of chars : 3
String length of literals : 8
String concatenation : BMS COLLEGE

3) toString() method : 242 Sanketh

5) String using getBytes() :
87 101 108 99 111 109 101
String using toCharArray() :
W e l c o m e

4) String using getChars() :
B M S CE

5) Bmsce equals Bmsce : true
Bmsce equals College : false
Bmsce equals BMSCE : false
Bmsce equals BMSCE : true

7) Substring is matched

8) String using startsWith() : true
String using startsWith() : false

9) strings using endswith() : true
Strings using endswith() : false

10) Hello equals Hello : true
Hello == Hello : false

11) apple ball cat dog ent free gun hen ice jug kite lift man net orange parrot queen ring star tree umbrella van watch xmas yatch zee

12) 1 2 3 4 5 6 7 8 9

13) This is a test. This is, too

14) Hellows world

15) Commege

16) Hello friends

17) student 1
name : Sanketh
Reg no : 242
Sem : 3
CGPA : 9.1
student 2
name : vayu
Reg no : 249
Sem : 3
CGPA : 9.8

18) String using charAt() : char at 3 is 'm'
String using getChars() : BMSCE
String reverse() : ECSMB

19) Eagle is flying
Eagle makes a sound

Hawk is flying
Hawk makes a sound

20) Circle_area = 314
Circle_peri = 62.8

Tri_area = 25
Tri_peri = 11

Sooleeth M Hanasi 1BM22CS242

16/01/24 Generics

1. Write a java program to create a generic class stack which hold 5 integers & 5 double values.

```
import java.util.*;

class Stack<E>{
    E stk[];
    int top;
    int size=10;

    Stack(){
        stk= (E[]) new Object[size];
        top=-1;
    }

    void push(E item){
        if(top==size-1)
            System.out.println("Overflow");
        else
            stk[++top]=item;
    }

    E pop(){
        if(top<0){
            System.out.println("Underflow");
            return null;
        }
        else{
            return stk[top--];
        }
    }
}
```

```
public class TestStack {
    public static void main (String [] args){
    Stack < Integer > mystack1 = new Stack< Integer>();
    Stack <Double.> mystack2 = new Stack <Double.>();

    Scanner s = new Scanner(System.in);
    System.out.println ("Enter Elements into the integer stack");
    For (int i=0; i<5; i++){
        Int n = s.nextInt();
        mystack1.push (n);
    }
    System.out.println("Enter Elements into the Double. Stack");
    for (int i=0 ; i< 5 ;i++){
        double m = s. nextDouble();
        mystack2.push (m);
    }
    System.out.println ("Elements of mystack1 ");
    for(int i=0 ; i<5 ;i++){
            System.out.println (mystack1.pop());
    }
    System.out.println ("Elements of mystack2 ");
    For (int i=0 ; i<5;i++) {
            System.out.println(mystack2.pop());
    }
    s.close();
}
}
```

Output:

Enter Elements into the Integer Stack

1 2 3 4 5

Enter Elements into the Double Stack

6 7 8 9 10

Elements of mystack1

1 2 3 4 5

Elements of mystack2

6.0 7.0 8.0 9.0 10.0

Sanketh M Hanasi 1BM22CS242