(8)
```
class Bmsc implements Runnable {
    String name;
    Thread t
    Bmsc (String threadName) {
        this.name = threadName;
        t = new Thread (this, this.nam
        t.start();
    }

    synchronized public void run(
        for(int i=0; i<=10; i++) {
            System.out.println("BMS College
            try {
                t.sleep(10000);
            }
            catch (InterruptedException e) {
                printStackTrace(e);
            }
        }
    }
}

class Cse implements Runnable {
    String name;
    Thread t;
    Cse(String threadName) {
        this.name = threadName;
        t = new Thread (this, this.name
        t.start();
    }
```

```
    }
  }
 }
}
class public class Thread1 {
  public static void main (String[] arg
    Bmsc bms = new Bmsc("BMS");
    Cse cs = new Cse("CS");
    bms.t.start();
    cs.t.start();
  }
}
```

Algorithm:

step 1: Start

step 2: Create new thread bms of type Bm
threadName = "BMS".

step 3: Declare String variable name & initializ
step hold the value of the passed th
parameter threadName.

step 4: create a new Thread object t
passing bms & bms.name as th
ters.

Step 5: Create a new Thread objec cs of
with threadName = "CS";

step 6: Declare String variable name &

step11: try Sleep(10000)
step 12: If InterruptedException is thrown.
error message.
step 13: start thread cse
step 14: for(int i=0; i<= 30; i++){
print "CSE".
step 15: If InterruptedException is thro
error message.
step 16: After the loop conditions fai
loops & stop the ~~string~~ Threa
step 17: stop.

Output:

BMS college of engineering
CSE
CSE
CSE
CSE
CSE
BMS college of engineering
CSE
CSE
CSE
CSE
CSE
BMS college of engineering.

```
1BM22CS024 Agneya D A
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
```