10)DEMONSTRATE INTER PROCESS COMMUNICATION AND DEADLOCK.

```
  class A {
   synchronized void foo(B b) {
      String name = Thread.currentThread().getName();
      System.out.println(name + " entered A.foo");

      try {
         Thread.sleep(1000);
      } catch (Exception e) {
         System.out.println("A Interrupted");
      }

      System.out.println(name + " trying to call B.last()");
      b.last();
   }

   void last() {
      System.out.println("Inside A.last");
   }
}

class B {
   synchronized void bar(A a) {
      String name = Thread.currentThread().getName();
      System.out.println(name + " entered B.bar");

      try {
         Thread.sleep(1000);
      } catch (Exception e) {
         System.out.println("B Interrupted");
      }
```

```
        System.out.println(name + " trying to call A.last()");
        a.last();
    }

    void last() {
        System.out.println("Inside B.last");
    }
}

class Deadlock implements Runnable {
    A a = new A();
    B b = new B();

    Deadlock() {
        Thread.currentThread().setName("MainThread");
        Thread t = new Thread(this, "RacingThread");

        t.start();



        a.foo(b);

        System.out.println("Back in main thread");
    }

    public void run() {

        b.bar(a);

        System.out.println("Back in other thread");
```

```
    }

    public static void main(String args[]) {

        System.out.println("Name: Aarusha GP, USN: 1BM23CS005");


        new Deadlock();
    }
}
```

OUTPUT: