import java.util.Scanner;

public class Bully {

class Pro {

int id;

boolean act;

Pro(int id)

{

this.id = id;

act = true;

}

}

int TotalProcess;

Pro[] process;

public Bully() { }

public void initialiseBully()

{

System.out.println("No of processes 5");

TotalProcess = 5;

process = new Pro[TotalProcess];

int i = 0;

while (i < process.length) {

process[i] = new Pro(i);

i++;

}

}

public void Election()

{

System.out.println("Process no "

+ process[FetchMaximum()].id

+ " fails");

process[FetchMaximum()].act = false;

System.out.println("Election Initiated by 2");

int initializedProcess = 2;

int old = initializedProcess;

int newer = old + 1;

while (true) {

if (process[newer].act) {

System.out.println(

"Process " + process[old].id

+ " pass Election(" + process[old].id

+ ") to" + process[newer].id);

old = newer;

}

newer = (newer + 1) % TotalProcess;

if (newer == initializedProcess) {

break;

}

}

System.out.println("Process "

+ process[FetchMaximum()].id

+ " becomes coordinator");

int coord = process[FetchMaximum()].id;

old = coord;

newer = (old + 1) % TotalProcess;

while (true) {

if (process[newer].act) {

System.out.println(

"Process " + process[old].id

+ " pass Coordinator(" + coord

+ ") message to process "

+ process[newer].id);

old = newer;

}

newer = (newer + 1) % TotalProcess;

if (newer == coord) {

System.out.println("End Of Election ");

break;

}

}

}

public int FetchMaximum()

{

int Ind = 0;

int maxId = -9999;

int i = 0;

while (i < process.length) {

if (process[i].act && process[i].id > maxId) {

maxId = process[i].id;

Ind = i;

}

i++;

}

return Ind;

}

public static void main(String arg[])

{

Bully object = new Bully();

object.initialiseBully();

object.Election();

}

}

