LAB 7

Write a C program to simulate deadlock detection.

16-8-21	
lab - 7 but duedlock d.t.	
1) Write a c program to structe deadlock detection	
# include 5 stdio, 45	V
Holding MAX pewers 10	4
# define MAX-Resources 10	
The pupeer; amounts;	
into Pallocations [MAX. quocen][MAX. Rnowners);	
int max. need [MAX-puscers) [MAX- Resources]:	
int ravailable [MAY_Resources];	
int marked [MAXProcess];	
int finished [MAX. Process];	
Nord initialise () ?	
punty ("Enter the no. of puocen: ");	
Scan ("1.d" & suscen)	
peint (Enter the no of opposite as ")	
10 gresources)	
the allocation to in it.	
TOT (120) Chunch : 111)	
for (j=0; j < resources ; ++) { scanf ("/.d"; & allocate (D(J)); } printl ("5to 10	
tor (j=0) (arrounds 1114) 5	
2 Scan ("/d" & allvot (DCO)	
3	
plent (Enter the max need	
for (1=0) 17 process (malaux In")	
penally ("Enter the max need making In"); } P (:	
for (j=0; < novumes: j++) searly ("1.2") max [i)[j];	
searl ("1. 2" " sit 4)	
2 mar Listis	
point ("Enter the and)	
for (i=0; i < ro. hypomes 1)	
1 ++) (n))	1
parity ("ENT -the available resources: In"); Scary ("/, d", & available T:2)	
a savaitable riss	

~	Poge ()
	Void datab databases
	Kold delit diedlock (?)
	for (i=0) ixproun ; +3)
	1
	marked (i)=0;
	finished [1)=0;
	int makel- count = 0;
	usule (markel - count < persons)
	1 (1 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
	9 (1 Sinshed [i) \$ \$! marked [i])
-	Achiet de Marched
	int can-allocate = 1;
	for (j=0; j< resources : j++)
	2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
	1) (man _ need (1)(j) _ allocation (1)(j) > allacation (1)
	1) (max_need li)[j)_allocation [i)[j) > available [i] can_allocate = 0)
	can audate -0)
	break; a the collection of the
	(Can-allocate)
	marked [i] = 1)
	marked court 1 + it. has a see all
	found = 1;
	for (j=0') < resourch ; j+1)
	available [j] 1 = allocation [i] 1j);
	3
	breek; Addith dedland de
	peint (Dandlock detected !: \n''); for 1 = 0 : 1 < perocen : 1+4)
	Dought (Doublak Astatul 1 : (n'');
	le Visa i Comment 11st)
	tor 11/0 1 proces 11-1-1

The state of the s	
4 (! fin hed [1] ? ! marked li) ? b pury ("process 1. d In", i); 3	
1 (! hin hed [137] 1 d In", i);	~ \
3 pury production	دلما
3	eo
entum;	a) b)
3 3 Cu + O Clark detailed in ");	0
perint ("No deadlock detailed in");	
int main ()	U
{ initialize ();	· ·
deliet deadlock ();	
3 sulum () Mar Mar Mar Mar Mar	
100 150 man (44)	
(1) Content: (1) (1) (1) (1) (1) (1) (1)	
Enter the no. of process: 5	
THE THE MAN AND THE	
Enter the allocation matrix: 0 10 and	
200	
(al 300 20) 11	
211	18 11 11
Enter the man great matrix: 17:53	3678
man new matrix: T.53	
3 2 2	
9 0 9	
222	
Enlin the available oursums: 2.	
No deadle de detait 1.	
Enlin the available ouround: 33	
Non	

OUTPUT:

C:\Users\Admin\Desktop\bm21cs065\deadlock_deec\bin\Debug\deadlock_deec.exe

```
Enter number of processes and number of resources required

3
Enter total number of required resources 5 for each process

7 5 3

3 2 2

9 0 2

2 2 2

4 3 3
Enter number of allocated resources 5 for each process

0 1 0

2 0 0

3 0 2

2 1 1

0 0 2

Enter number of available resources

1 1 1

Deadlock occurred

Process returned 0 (0x0) execution time: 65.375 s

Press any key to continue.
```

C:\Users\Admin\Desktop\bm21cs065\deadlock_deec\bin\Debug\deadlock_deec.exe Enter number of processes and number of resources required 5 3 Enter total number of required resources 5 for each process 7 5 3 3 2 2 9 0 2 2 2 2 4 3 3 Enter number of allocated resources 5 for each process 0 1 0 200 3 0 2 2 1 1 0 0 2 Enter number of available resources 3 3 2 No deadlock Process returned 0 (0x0) execution time : 86.778 s Press any key to continue.