

Deadlock

Recall: Deadlock

- Situation in which group of threads/processes all block forever
- Typically, each holds a resource that others are blocking on



<http://minutillo.com/steve/weblog/2003/1/21/deadlock/>,
where it is attributed to "Chuck @ China"
(<http://chake.chinatefl.com/>)

Modeling resource contention deadlocks

- Focus on requesting and freeing resources
- Assume process blocks if requests something in use
- Vertices for each process and each resource
- Edge from resource to process holding it, from process to resource it is blocking on

- Example:

Process A	Process B	Process C
request R	request S	request T
request S	request T	request R
free R	free S	free T
free S	free T	free R

Detecting deadlock

- Use DFS to look for cycle in graph
- Vertex colors: “unvisited”, “in progress”, “done”

all vertices initially “unvisited”

for each “unvisited” vertex v : visit(v);

visit(vertex v):

 color v “in progress”;

 for each neighbor u of v :

 if(u is “in progress”) print “found cycle”;

 if(u is “unvisited”) visit(u);

 color v “done”;

Detection strategy w/ multiple copies of resources

(using vectors and matrices...)

Is the following situation safe?

additional resources needed per job

R =

1	0	2
2	2	1
0	1	1
3	1	0

currently used resources per job

C =

1	1	0
0	1	1
2	0	0
1	0	0

currently available resources

A =

2	1	1
---	---	---

A. Yes

B. Yes

C. No

D. No

E. As long as you don't call on me to explain my answer

Is the following situation safe?

additional resources needed per job

R =

1	0	2
2	2	1
0	1	1
3	1	0

currently used resources per job

C =

1	1	0
0	1	1
2	0	0
1	0	0

currently available resources

A =

2	1	1
---	---	---

A. Yes

B. Yes

C. No

D. No

E. As long as you don't call on me to explain my answer

Strategy 1: Deadlock avoidance ("Banker's Algorithm")

- Eliminate possibility of deadlock through clever resource allocation
- For each resource request, evaluate if granting it puts system into an unsafe state (only grant it if not)

Would the Banker's Algorithm grant a request from job 2 for 1 unit of resource 3 in the following situation?

currently available resources

A =

2	1	1
---	---	---

currently used resources per job

C =

1	0	2
2	2	1
0	1	1
0	1	0

additional resources needed per job

R =

1	0	1
3	2	1
0	2	0
2	1	0

A. Yes

B. No

C. None of the above

Would the Banker's Algorithm grant a request from job 2 for 1 unit of resource 3 in the following situation?

currently available resources

A =

2	1	1
---	---	---

currently used resources per job

C =

1	0	2
2	2	1
0	1	1
0	1	0

additional resources needed per job

R =

1	0	1
3	2	1
0	2	0
2	1	0

A. Yes

B. No

C. None of the above