Chrysilla Citra Windyadari
21083010023

No.

Date

Sistem Operasi B

Kondisi - kondisi pemicu deadlock :

1) Mutual exclusion (mutual exclusion conditional) : kondisi ketika suatu proses menggunakan resource tertentu maka tidak diperbolehkan proses lain menggunakan resource tersebut. Contoh : hanya satu mobil boleh menempati setiap persimpangan pada suatu waktu.
2.) Kondisi genggam dan tunggu (hold and wait) : pada saat suatu proses mengakses suatu resource, proses tersebut dapat meminta ijin untuk mengakses resource lain. Contoh : mobil boleh diam di persimpangan ketika menunggu untuk sampai ke persimpangan berikutnya
3.) Kondisi non-preemption (non-preemption condition) : jika suatu proses meminta ijin untuk mengakses resource, sementara resource tidak tersedia, maka permintaan tidak dapat dibatalkan. Contoh : mobil tidak dapat dipindahkan dari tempatnya pada arus lalu lintas, hanya dapat jalan ke depan
4.) Kondisi menunggu secara sirkuler (circular wait condition) : jika proses Pi sedang mengakses resource Ri, dan meminta ijin untuk mengakses resource Rj, dan pada saat bersamaan proses Pj sedang mengakses Rj dan minta ijin untuk mengakses resource Ri. Contoh : kumpulan mobil dalam situasi deadlock termasuk juga mobil yang ada ditengah persimpangan.

Penanganan deadlock

1) Mengabaikan permasalahan (the ostrich Algorithm) : jika terjadi deadlock untuk tipe penanganan ini maka sistem operasi akan menanggulangi dengan cara tidak mendeteksi deadlock dan membiarkannya secara otomatis menahikan program seakan-akan tidak terjadi apa pun.
2) Deteksi dan pemulihan (recovery) : mengijinkan sistem mengalami deadlock agar dapat dideteksi kemudian sistem tersebut segera diperbaiki.
3.) Pencegahan dengan menradakan salah satu dari empat kondisi deadlock :
   - Meniadakan mutual exclusion : melakukan spooling perangkat - perangkat yang harus didedikasikan ke suatu proses. Dengan spooling permintaan - permintaan diantrikan di harddisk.
   - Meniadakan hold and wait : dengan mengalokasikan semua sumber daya atau tidak sama sekali dan juga dengan hold and release
   - Meniadakan non - preemption : jika suatu proses yang membawa beberapa resource meminta resource lain yang tidak dapat segera dipenuhi maka semua resource yang sedang dibawa proses tersebut harus dibebaskan ; proses yang sedang dalam keadaan menunggu maka resource yang dibawanya dihunda dan ditambahkan pada daftar resource

No.

Date . .

- Meniadakan circular wait : proses hanya diperbolehkan menggenggam satu resource

4.) Pengalokasian sumber daya yang efisien : situasi ketika resource dialokasikan pada penggunaan nilai tertinggi, tidak ada alternatif untuk menggunakan resource lebih lanjut tanpa membuat yang lain lebih buruk