

Deadlock adalah keadaan dimana sejumlah permintaan yang tidak bisa dijalankan oleh scheduler karena permintaan-permintaan tersebut saling tunggu menunggu.

Kondisi Untuk Mencapai Deadlock

1. Mutual exclusion conditional

Jika suatu proses menggunakan suatu resource, tidak ada proses lain yang boleh menggunakan resource tersebut.

2. Hold and wait

Pada saat suatu proses mengakses suatu resource, proses tersebut dapat meminta izin untuk mengakses resource lain.

3. No-Preemption condition

Jika suatu proses meminta izin untuk mengakses resource, sementara resource tidak tersedia, maka permintaan tidak dapat dibatalkan.

4. Circular wait condition

Jika proses P_i sedang mengakses resource R_i , dan meminta izin untuk mengakses resource R_j . dan pada saat bersamaan proses P_j sedang mengakses R_j dan meminta izin untuk mengakses resource R_i .

Penanganan Deadlock

1. Mengabaikan permasalahan (The Ostrich Algorithm)

Dalam algoritma ini ditetapkan bahwa untuk menghadapi deadlock ialah dengan berpura-pura bahwa tidak ada masalah apapun.

2. Deteksi dan Pemulihan (recovery)

Mendeteksi sistem mana yang terlibat di dalamnya. Setelah diketahui sistem mana saja yang terlibat maka diadakan proses untuk memperbaiki dan menjadikan sistem berjalan kembali.

3. Pencegahan, dengan meniadakan salah satu dari empat kondisi deadlock

Meniadakan mutual exclusion \rightarrow melakukan spooling perangkat² yg harus didedikasikan ke suatu proses. By spooling, permintaan² diantrikan di harddisk

4. Pengalokasian sumber daya yang efisien.

Jika semua sumber daya tersedia, proses dialokasikan yang diperlukan dan berjalan sampai selesai. Jika tidak tersedia harus menunggu sampai semua sumber daya yg diperlukan tersedia untuk dialokasikan.