

Kondisi untuk mencapai Deadlock.

1. Mutual exclusion (mutual exclusion conditional)

↳ keadaan atau sebuah kondisi dimana hanya ada satu proses yang dapat mengakses atau menggunakan sumber daya

↳ hanya ada satu proses pada satu waktu yang diperbolehkan untuk menjalankan sebuah perintah.

2. Kondisi genggam dan tunggu (hold and wait)

↳ Proses yang sedang menunggu sumber daya, dan dapat meminta lagi apabila sumber daya pertama sudah benar-benar tidak terpakai lagi.

3. Non - preemption (non - preemption condition)

↳ Proses pembatalan suatu perintah apabila perintah tidak tersedia. Sebelumnya harus memiliki suatu ijin untuk mengakses perintah tsb.

4. Circular wait (Circular wait condition)

↳ Terdapat sekumpulan proses (P_0, P_1, \dots, P_n) yang menunggu R, dimana P_0 menunggu R yang dibawa P_1 , P_1 menunggu R yang dibawa P_2 dst.

Penanganan Deadlock

1. Mengabaikan Permasalahan (The Ostrich Algorithm)

↳ Strategi mengabaikan masalah yang mungkin terjadi atas dasar bahwa masalah itu mungkin sangat jarang terjadi - "Menempel di kepala pasir dan berharap" bahwa tidak terjadi masalah"

2. Deteksi dan Pemulihan (Recovery)

↳ Memeriksa apakah telah terjadi deadlock dan menentukan proses^{xx} dan sumber daya^{xx} yg terlibat deadlock secara presisi.

3. Pencegahan, dengan meniadakan salah satu dari 4 kondisi deadlock

↳ Meniadakan Mutual exclusion dengan cara spooling peralatan^{xx} yg harus di dialokasikan untuk suatu proses

4. Pengalokasian sumber daya yang efisien.

↳ Jika sumber daya tersedia, proses dialokasikan yang diperlukan dan berjalan sampai selesai.

