

Sistem Operasi

Pertemuan 07

Alauddin Maulana Hirzan, S.Kom., M.Kom.

NIDN. 0607069401

Fakultas Teknologi Informasi dan Komunikasi, Universitas Semarang



Inter-Process Communication

Jenis Message Passing

Message Passing memiliki dua jenis cara komunikasi:

① Komunikasi Sinkron:

- Dalam komunikasi sinkron, pengirim dan penerima disinkronkan dalam waktu.
- Pengirim memblokir hingga pesan diterima dan diproses oleh penerima.

② Komunikasi Asinkron:

- Dalam komunikasi asynchronous, pengirim dan penerima beroperasi secara independen satu sama lain dalam hal waktu.
- Pengirim tidak menunggu penerima memproses pesannya segera setelah dikirim.

Inter-Process Communication

Mekanisme Message Passing - Pipes

Definisi:

Dalam bidang sistem operasi dan ilmu komputer, pipa merupakan salah satu bentuk komunikasi antar proses (IPC) yang memfasilitasi transfer data antara dua proses. Ini bertindak sebagai saluran di mana output dari satu proses dapat diumpangkan secara langsung sebagai input ke proses lain tanpa memerlukan file sementara. Pipa memungkinkan proses untuk berkomunikasi dan menyinkronkan aktivitas mereka secara efisien.

Inter-Process Communication

Mekanisme Message Passing - Pipes

Karakteristik:

- 1 **Komunikasi Searah:** Pipa memfasilitasi komunikasi satu arah antar proses.
- 2 **First-In-First-Out (FIFO):** Pipa beroperasi dengan basis FIFO, artinya data yang ditulis ke dalam pipa terlebih dahulu akan dibaca terlebih dahulu.
- 3 **Ukuran Buffer Terbatas:** Pipa biasanya memiliki ukuran buffer yang terbatas.
- 4 **Komunikasi Anonim:** Pipa dapat tidak diberi nama atau diberi nama.

Inter-Process Communication

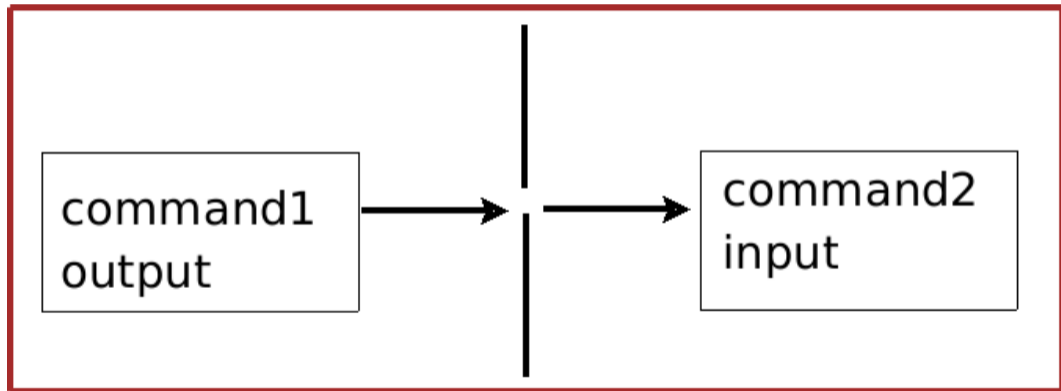
Mekanisme Message Passing - Pipes

Jenis:

- 1 **Nameless Pipes:** Juga dikenal sebagai pipa anonim, pipa ini dibuat oleh sistem operasi dan biasanya digunakan untuk komunikasi antara proses induk dan anak atau antara proses yang dibuat oleh nenek moyang yang sama.
- 2 **Named Pipes:** Juga dikenal sebagai FIFO (First-In-First-Out), pipa bernama adalah file khusus yang dibuat dalam sistem file yang bertindak sebagai saluran komunikasi antara proses yang tidak terkait.

Inter-Process Communication

Mekanisme Message Passing - Pipes



Inter-Process Communication

Mekanisme Message Passing - Socket

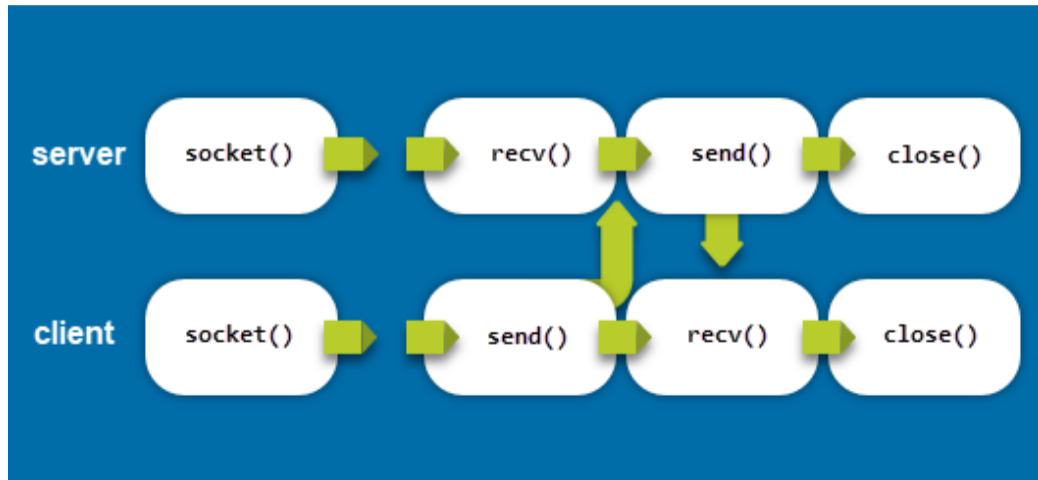
Definisi:

Socket adalah titik akhir untuk komunikasi antara dua mesin atau proses melalui jaringan. Mereka memungkinkan aliran data dua arah antar proses, baik pada mesin yang sama atau melalui jaringan. Pada dasarnya, socket mewakili satu titik akhir dari suatu tautan komunikasi.

Socket beroperasi berdasarkan model klien-server, di mana satu sisi komunikasi bertindak sebagai server, mendengarkan koneksi masuk, dan sisi lain bertindak sebagai klien, memulai koneksi ke server. Setelah koneksi dibuat, data dapat dikirim antara klien dan server.

Inter-Process Communication

Mekanisme Message Passing - Socket



Inter-Process Communication

Mekanisme Message Passing - Message Queue

Definisi:

Antrean pesan adalah struktur data yang menyimpan pesan, memungkinkan komunikasi antara berbagai bagian sistem. Mereka mengikuti prinsip FIFO (Masuk Pertama, Keluar Pertama), artinya pesan pertama yang ditambahkan ke antrean adalah pesan pertama yang diproses. Antrean ini memisahkan pembuat pesan dari konsumennya, sehingga mereka dapat beroperasi secara independen dan sesuai kecepatan mereka sendiri.

Inter-Process Communication

Mekanisme Message Passing - Message Queue



Inter-Process Communication

Mekanisme Shared Memory

Definisi:

Shared Memory adalah mekanisme yang memungkinkan beberapa proses atau thread mengakses struktur data umum secara bersamaan. Dalam paradigma ini, wilayah memori ditetapkan sebagai milik bersama, sehingga memungkinkan berbagai proses untuk membaca dan menulis ke lokasi memori yang sama.

Ruang **Shared Memory** ini biasanya berada di RAM sistem dan dapat diakses oleh semua proses yang berpartisipasi.

Inter-Process Communication

Mekanisme Shared Memory

Dalam sistem memori bersama, proses dapat berkomunikasi baik dengan membaca dan menulis langsung ke variabel bersama atau dengan menggunakan sinkronisasi primitif seperti semaphore, mutex, atau monitor untuk mengoordinasikan akses ke sumber daya bersama.

Inter-Process Communication

Mekanisme Shared Memory - Semaphores

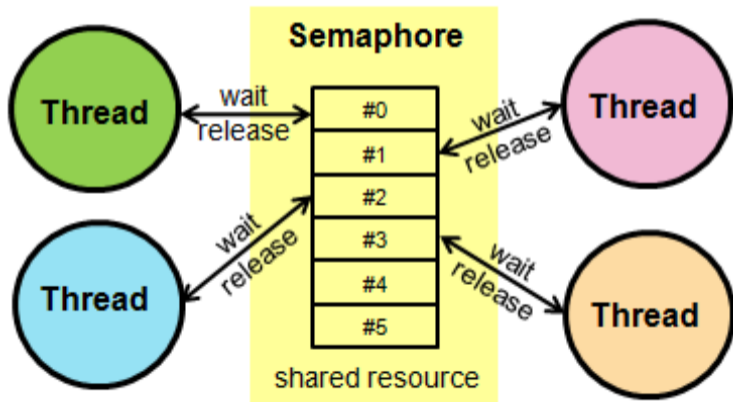
Definisi:

Semaphore adalah sinkronisasi primitif yang digunakan dalam pemrograman bersamaan untuk mengontrol akses ke sumber daya bersama oleh beberapa proses atau thread. Mereka diperkenalkan oleh **Edsger W. Dijkstra** pada tahun 1965 dan sejak itu menjadi konsep dasar dalam sistem operasi dan pemrograman bersamaan.

Inter-Process Communication

Mekanisme Shared Memory - Semaphores

Mekanism:



Inter-Process Communication

Mekanisme Shared Memory - Mutexes

Defisi:

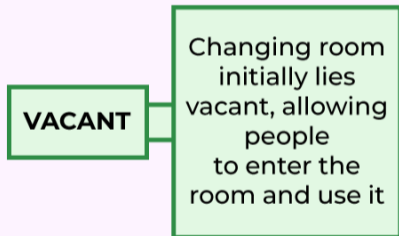
Mutex, kependekan dari objek pengecualian bersama, adalah primitif sinkronisasi yang digunakan dalam pemrograman bersamaan untuk mencegah beberapa thread mengakses sumber daya bersama secara bersamaan seperti variabel, struktur data, atau bagian kode yang tidak boleh dieksekusi secara bersamaan.

Mutex memungkinkan thread untuk mengoordinasikan akses ke sumber daya bersama secara terkendali, memastikan bahwa hanya satu thread yang dapat mengakses sumber daya pada waktu tertentu.

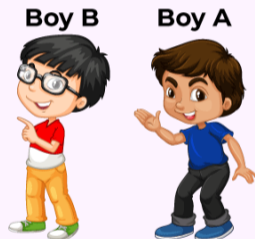
Inter-Process Communication

Mekanisme Shared Memory - Mutexes

Mekanisme:



Since Boy A is done using the changing room, he vacates it-making the sign outside the room change.



Since the changing room has been vacated by Boy A, Boy B can enter the hanging room.

Inter-Process Communication

Mekanisme Shared Memory - Signal

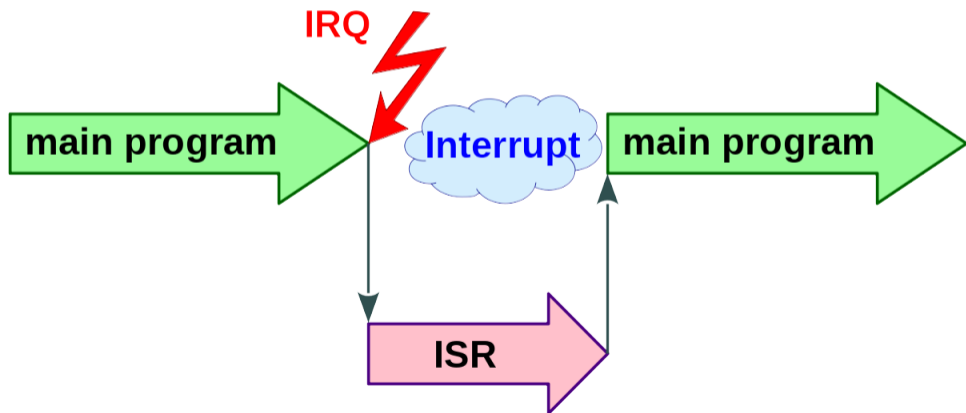
Definisi:

Sinyal dalam komputasi dapat didefinisikan sebagai interupsi perangkat lunak yang dikirimkan ke suatu proses, memberitahukannya tentang suatu peristiwa yang terjadi dalam sistem. Peristiwa ini dapat berkisar dari interupsi perangkat keras (seperti pengguna menekan Ctrl+C untuk menghentikan suatu proses) hingga kondisi perangkat lunak (seperti proses yang mencoba mengakses alamat memori yang tidak valid).

Inter-Process Communication

Mekanisme Shared Memory - Signal

Mekanisme:



Inter-Process Communication

Shared Memory vs Message Passing

1. Shared Memory:

- ① **Kinerja:** Memori bersama dapat menawarkan kinerja tinggi karena proses dapat mengakses data bersama secara langsung
- ② **Sinkronisasi:** Mekanisme sinkronisasi seperti mutex, semaphore, atau monitor diperlukan untuk mengoordinasikan akses
- ③ **Kompleksitas:** Penerapan komunikasi memori bersama memerlukan manajemen sinkronisasi yang cermat
- ④ **Model Pemrograman:** Shared Memory seringkali lebih mudah diprogram

Inter-Process Communication

Shared Memory vs Message Passing

2. Message Passing:

- 1 **Isolasi:** Proses memiliki ruang alamat terpisah, yang menyediakan isolasi dan perlindungan.
- 2 **Sinkronisasi:** Pengiriman pesan biasanya melibatkan sinkronisasi eksplisit melalui antrian pesan, saluran, atau lainnya
- 3 **Skalabilitas:** Pengiriman pesan dapat diskalakan dengan baik untuk sistem terdistribusi karena tidak memerlukan akses memori bersama
- 4 **Overhead:** Pengiriman pesan dapat menimbulkan overhead yang lebih tinggi dibandingkan dengan memori bersama karena kebutuhan untuk menyalin data

Inter-Process Communication

Penanggulangan Galat dan Perbaikan

1. Deteksi Galat

- 1 **Checksum dan CRC:** Menambahkan checksum ke pesan dapat mendeteksi kesalahan
- 2 **Nomor Urutan:** Menetapkan nomor urut ke pesan memungkinkan proses mendeteksi pesan yang hilang atau rusak,
- 3 **Batas waktu:** Menggunakan batas waktu untuk penerimaan atau respons pesan dapat mendeteksi kegagalan atau penundaan komunikasi.
- 4 **Kode Kesalahan:** Menyertakan kode kesalahan dalam pesan membantu mengidentifikasi jenis kesalahan tertentu, seperti data yang tidak valid atau pelanggaran protokol.

Inter-Process Communication

Penanggulangan Galat dan Perbaikan

2. Perbaikan Galat

- 1 **Transmisi Ulang:** Jika pesan hilang atau rusak, pengirim dapat mengirimkan ulang pesan tersebut untuk memastikan pengirimannya.
- 2 **Redundansi:** Data atau proses yang berlebihan dapat digunakan untuk menyediakan cadangan jika terjadi kegagalan
- 3 **Mekanisme Percobaan Ulang:** Menerapkan mekanisme percobaan ulang memungkinkan proses untuk mencoba kembali operasi yang gagal
- 4 **Rollback dan Kompensasi:** Dalam sistem transaksional, mekanisme rollback dan kompensasi dapat digunakan untuk mengembalikan atau mengkompensasi transaksi yang gagal

