

# Dukungan Sistem Operasi

## Pertemuan 11

Oleh :

Riyanto Sigit, ST, M.Kom

Nur Rosyid, ST

Setiawardhana, ST

Hero Yudo Martono, ST

# Tujuan

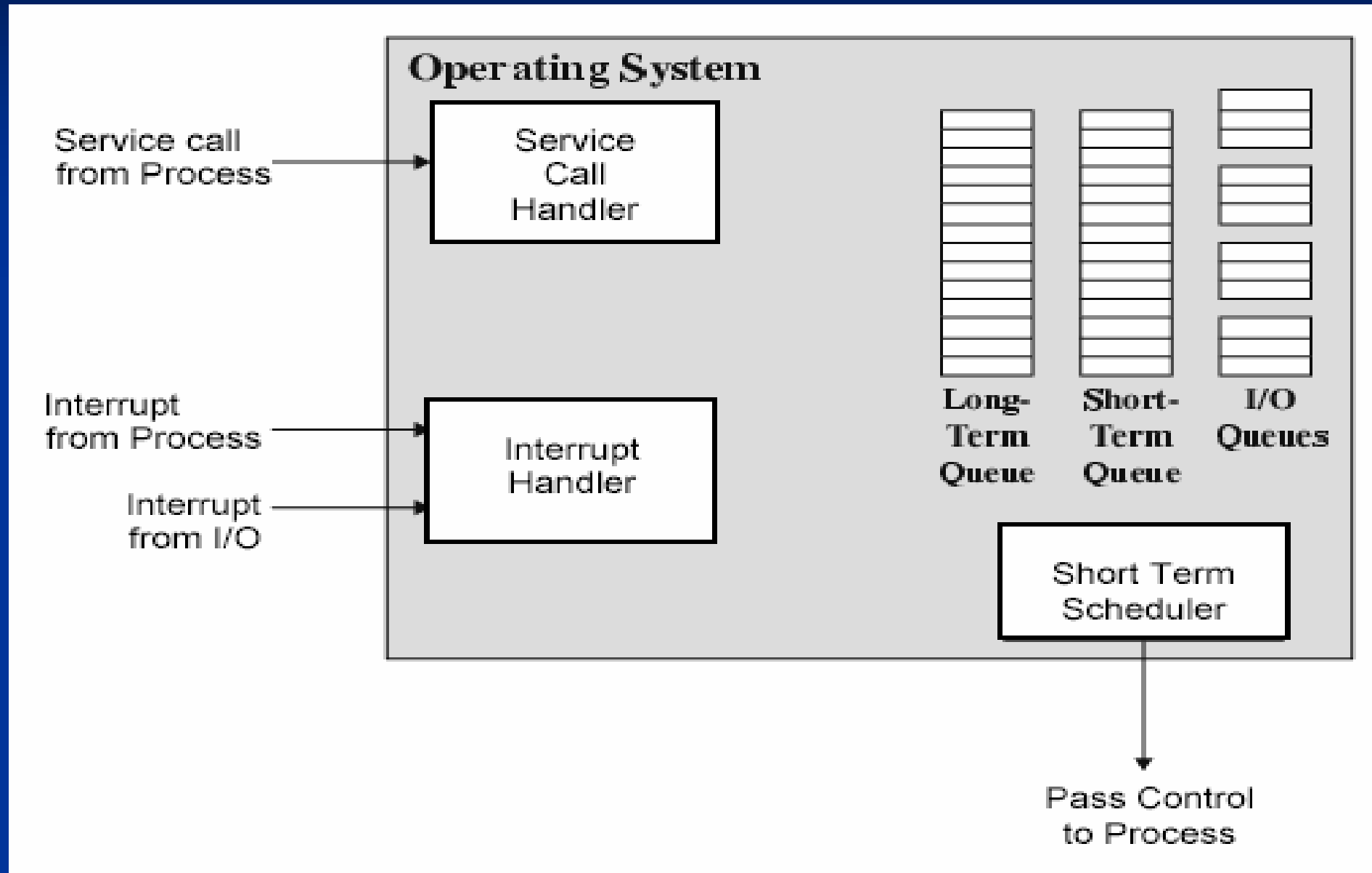
1. Memahami tentang system operasi
2. Memahami tentang penjadwalan proses
3. Memahami Jenis-jenis memori management  
Mengerti tentang Swaping dan Partition
4. Memahami konsep Paging, segmentation
5. Mengerti konsep Vitual memori.
6. Mengerti implemantasi pengaturan memori pada powerPC

# Representasi Proses dalam Sistem Operasi melalui Blok Kontrol Proses

<b>Identifier</b>
<b>State</b>
<b>Priority</b>
<b>Program counter</b>
<b>Memory pointers</b>
<b>Context data</b>
<b>I/O status information</b>
<b>Accounting information</b>
<b>• • •</b>

- ***Identifier*** : setiap proses saat itu memiliki identifikasi yang unik
- ***State*** : keadaan proses saat itu (new, ready, dsb)
- ***Priority*** : tingkatan proiritas relatif.
- ***Program Counter*** : alamat instruksi berikutnya di dalam program yang akan dieksekusi.
- ***Memori Pointers*** : lokasi awal dan akhir proses di dalam memori.
- ***Context Data*** : data yang berada di dalam register prosesor pada saat proses dieksekusi.
- ***I/O Status Information*** : meliputi request I/O yang belum dipenuhi, perangkat I/O yang terpakai oleh proses ini.
- ***Accounting Information*** : meliputi jumlah waktu prosesor dan clock time yang digunakan, batas waktu, nomer account.

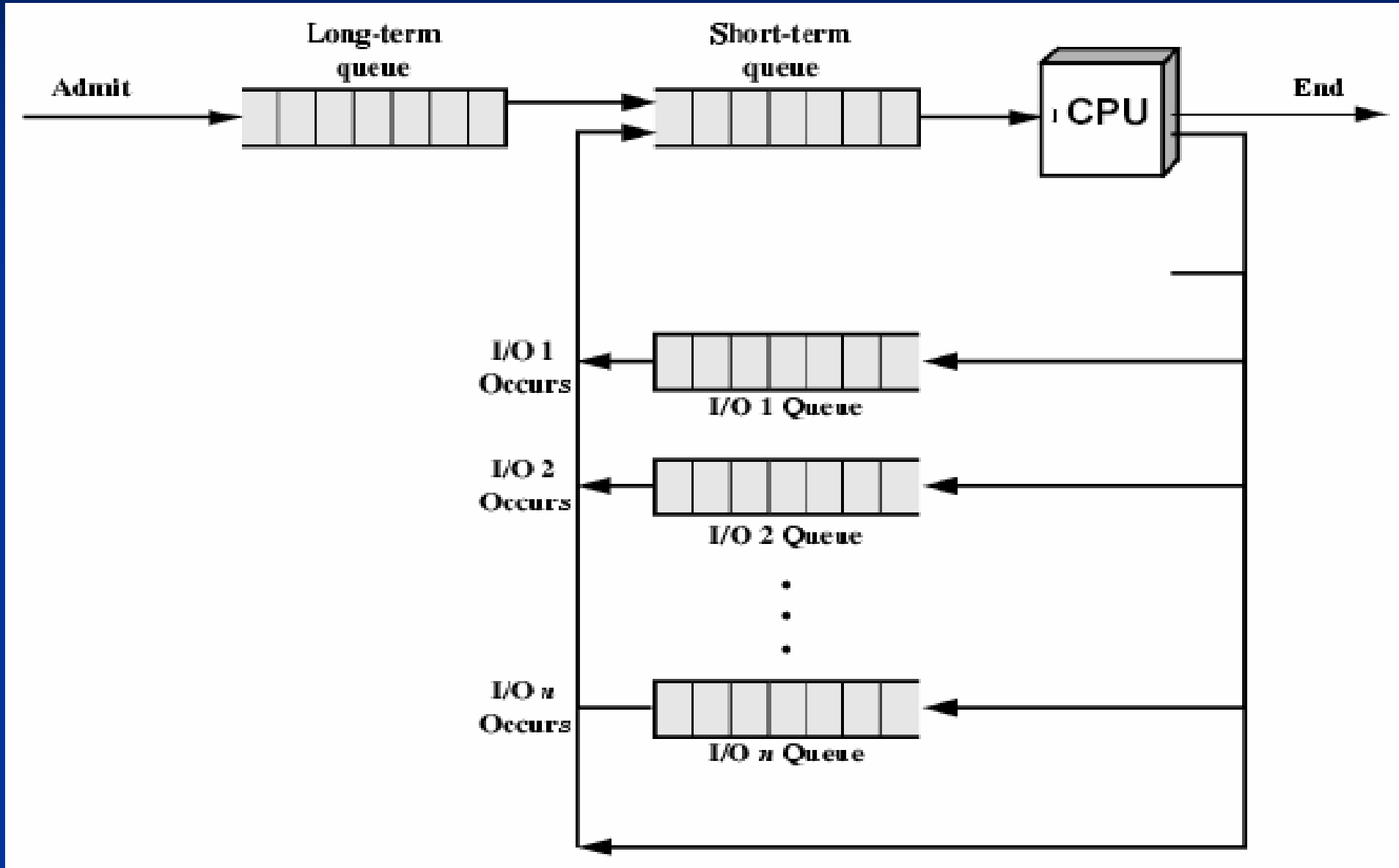
# Elemen-elemen Sistem Operasi



# Elemen dan Penjadwalan Proses

- Sistem operasi menerima kontrol prosesor pada interrupt handler
- Hal ini bila terjadi interupsi dan pada service-call handler bila terjadi panggilan layanan
- Sekali interupsi atau panggilan layanan ditangani maka penjadwal jangka pendek dipanggil untuk mengambil proses untuk selanjutnya diproses

# Representasi Diagram Antrian Penjadwalan



# Manajemen Memori

- Melakukan pengaturan prosesor sehubungan penjadwalan proses yang harus dilakukannya
- untuk mempersiapkan proses multi-programming diperlukan pengaturan memori.
- Dalam sistem single-programming, memori utama dibagi menjadi dua bagian, yaitu: satu bagian untuk sistem operasi (monitor resident) dan bagian lainnya untuk program yang sedang berjalan.
- Dalam sistem multi-programming, bagian pengguna dari memori harus dibagi lagi untuk mengakomodasikan sejumlah proses.
- Tugas pembagian dilakukan secara dinamis oleh sistem operasi yang dikenal dengan manajemen memori.
- Ada beberapa istilah dalam manajemen memori yang akan dijelaskan, yaitu: swapping, partitioning, paging, virtual memory.



# Swapping

- Kalau kita memperhatikan proses pada gambar (Representasi diagram antrian penjadwalan)
- Kerja prosesor lebih cepat dari pada perangkat I/O maka akan terjadi antrian keluaran, yaitu proses yang memerlukan akses ke perangkat I/O.
- Pendekatan untuk masalah diatas adalah swapping.
- Kita memiliki antrian panjang permintaan proses, yang umumnya tersimpan dalam memori utama.
- Apabila proses telah selesai, maka proses – proses tersebut akan dikeluarkan dari memori utama.
- Sekarang, situasi yang terjadi adalah tidak ada proses di dalam memori utama yang siap, daripada prosesor idle maka prosesor akan malakukan swap salah satu tersebut kembali ke disk di antrian menengah. Antrian ini merupakan antrian proses yang telah dikeluarkan sementara dari memori.

# Swapping

- Kemudian sistem operasi mengambil proses lain dari antrian menengah, atau memenuhi permintaan proses yang baru dari antrian panjang.
- Setelah itu eksekusi akan dilanjutkan dengan memproses proses yang baru tiba.
- Swapping merupakan operasi I/O, karenanya dapat membuat keadaan menjadi buruk, namun karena I/O disk umumnya lebih cepat dari I/O lainnya maka swapping ini mampu meningkatkan kinerja prosesor

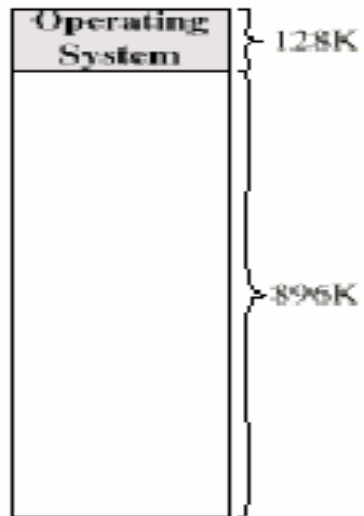
# Partitioning

- Sistem operasi akan menempati bagian memori yang tetap.
- Sisa memori dibagi – bagi untuk keperluan sejumlah proses.
- Partitioning adalah teknik membagi memori menjadi beberapa bagian sesuai kebutuhan.
- Terdapat dua macam partisi, yaitu partisi tetap (fixed size partitioning) dan partisi variabel (variable size partitioning).
- Dalam partisi tetap, proses akan disimpan dalam partisi memori yang dapat menampungnya.
- Partisi biasanya dibuat tidak sama kapasitasnya.
- Pendekatan ini kurang efisien karena tidak bisa diubah ukuran partisinya dan biasa terjadi sisa memori dari alokasi partisinya sehingga terjadi lobang (hole) memori

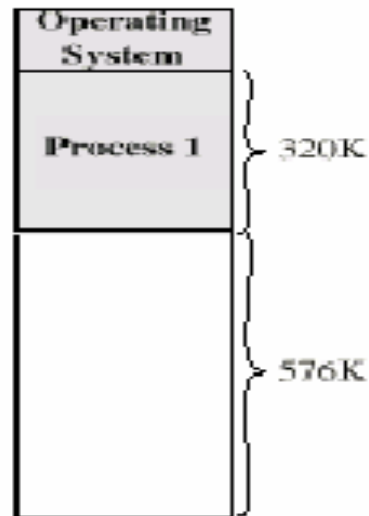
# Partitioning

- Pendekatan yang lebih efisien adalah partisi variabel, dimana ukuran partisi disesuaikan kebutuhan memorinya.
- Suatu blok memori akan diisi proses lain apabila tidak digunakan lagi oleh suatu proses.
- Akibat dari proses ini adalah terjadinya fragmentasi dari program atau proses yang dijalankan sehingga utilitas memori akan menurun.
- Teknik untuk mengatasi hal ini adalah compaction. Dari waktu ke waktu, sistem operasi memindahkan proses di dalam memori untuk menempatkan seluruh memori yang bebas secara bersama – sama pada sebuah blok.
- Dengan adanya pemindahan blok – blok proses maka akan terjadi perubahan alamat memori.
- Namun perlu diketahui bahwa terdapat dua macam alamat dalam hal ini, yaitu alamat fisik dan alamat logik. Alamat fisik adalah alamat sebenarnya pada memori, sedangkan alamat logik adalah alamat yang berhubungan dengan akses instruksi ke prosesor.

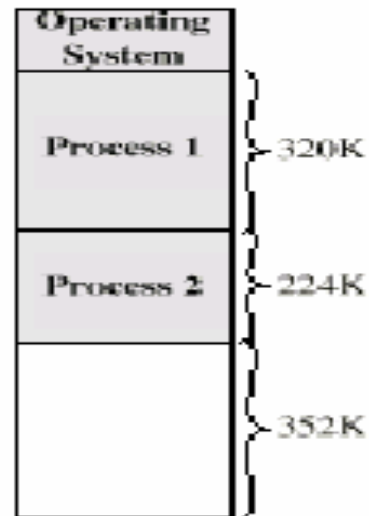
# Efek Partisi Variabel



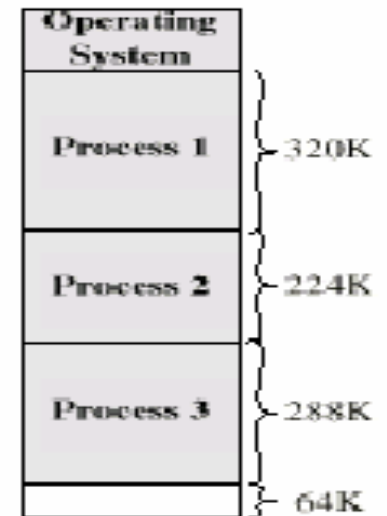
(a)



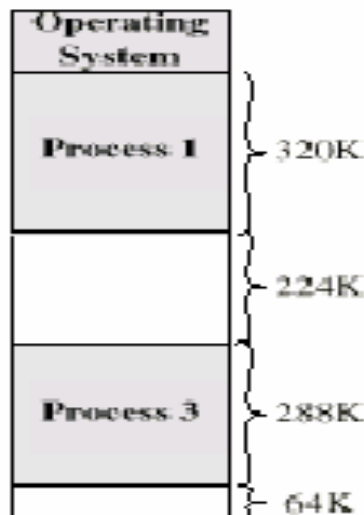
(b)



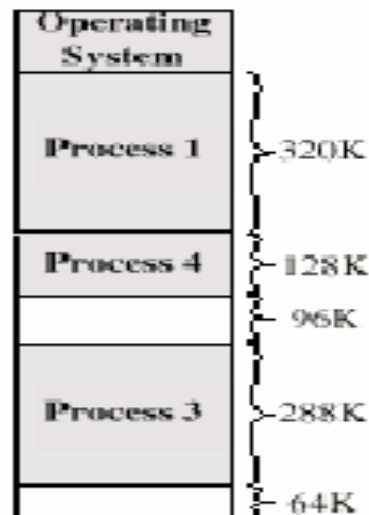
(c)



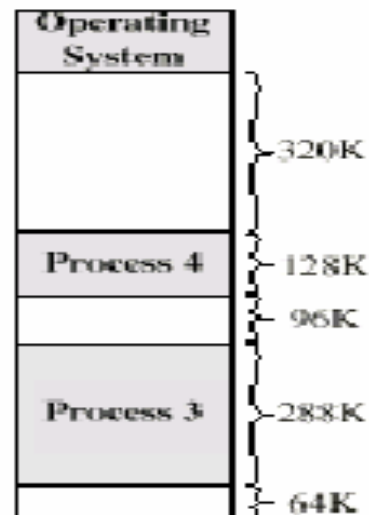
(d)



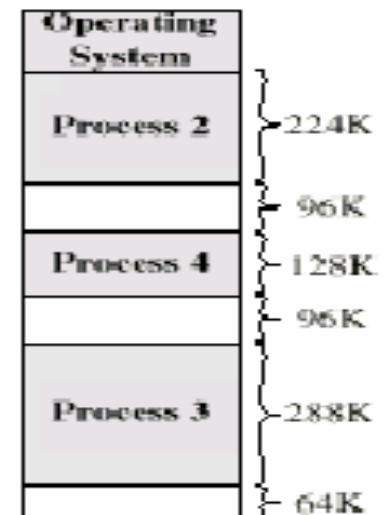
(e)



(f)



(g)

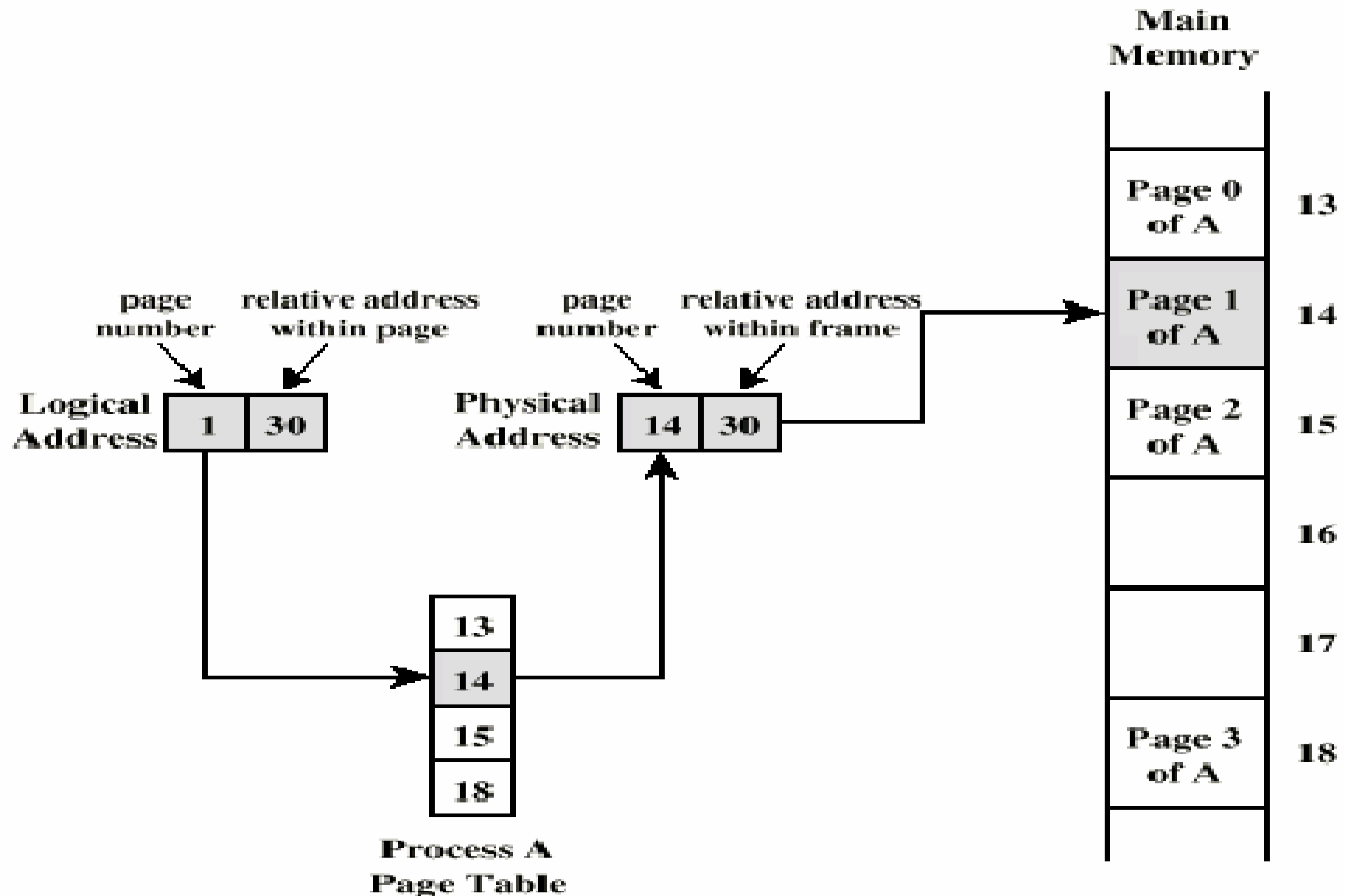


(h)

# Paging

- Penggunaan partisi tidak cukup efisien dalam penggunaan memori.
- Terdapat metode lain yang disebut paging.
- Paging adalah membagi memori utama menjadi frame – frame kecil berukuran sama.
- Setiap proses dibagi menjadi page – page berukuran sama dengan frame. Proses yang lebih kecil membutuhkan page yang lebih sedikit, sedangkan proses – proses besar akan memerlukan page yang lebih banyak.
- Ketika sebuah proses dibawa ke dalam memori, page – page-nya dimuatkan ke dalam frame yang tersedia, dan kemudian page tabel dibentuk. Page tabel digunakan untuk mengubah alamat logik menjadi alamat fisik dan juga sebaliknya.

# Proses Transformasi Alamat pada Paging Table



# Paging

- Pada saat suatu proses sedang berjalan, sebuah register menyimpan alamat awal page table proses tersebut.
- Nomer halaman alamat virtual digunakan untuk mengindeks tabel tersebut dan meng-look up nomer frame-nya. Hal ini dikombinasikan dengan bagian offset dari alamat virtual untuk menghasilkan alamat real yang diinginkan.
- Tabel ini bisa sangat besar, sehingga sebagian besar teknik ini menyimpan tabel pada memori virtual, bukan pada real memori.
- Pada saat suatu proses sedang berjalan, sedikitnya bagian page table-nya harus berada di memori utama, termasuk page table entry yang sedang dieksekusi.



# Paging

- Sebagian prosesor menggunakan teknik dua tingkat untuk mengorganisasikan page table yang besar.
- Dalam sistem ini, terdapat page directory sama dengan  $X$ , dan apabila panjangnya  $Y$ , maka sebuah proses dapat berisi sampai dengan  $X \times Y$  page.
- Umumnya panjang maksimum page table dibatasi sampai sama dengan satu page.
- Struktur inverted page table menerapkan teknik ini dan telah digunakan pada komputer IBM AS/400 dan pada semua produk RISC, termasuk PowerPC.
- Terdapat hash table yang berisi sebuah pointer ke page table yang diinversikan, yang berisi page table entry.
- Dengan struktur seperti ini maka hash table hanya akan terdapat sebuah entry dan page table yang diinversikan bagi setiap real memory page, bukannya satu per virtual page.



# Virtual Memory (1)

- Untuk memahami virtual memori perlu diawali dari pemahaman mendalam tentang paging.
- demand paging, yaitu hanya page – page dari proses yang dibutuhkan saat itu saja yang dimuatkan ke memori utama. Hal ini merupakan solusi bahwa memori utama biasanya berukuran kecil dan mahal.
- Apabila ada sebuah program besar maka akan tidak efektif apabila semua page dari program tersebut diletakkan dalam memori utama.
- Hanya page yang akan digunakan saja yang dimuatkan dalam memori utama membuat kinerja memori lebih baik.

# Virtual Memory (2)

- Hanya page yang akan digunakan saja yang dimuatkan dalam memori utama membuat kinerja memori lebih baik.
- Apabila page yang akan dieksekusi tidak didapatkan di memori utama, maka sinyal page fault diaktifkan.
- Sinyal ini menyatakan bahwa sistem operasi harus mengambil page yang dimaksud.
- Karena proses hanya mengeksekusi di dalam memori utama saja, maka memori tersebut disebut real memory.
- Namun pemrogram atau pengguna dapat menggunakan memori yang lebih besar, yang dikenal sebagai virtual memory.
- Memori virtual merupakan solusi efektif bagi pengguna atau pemrogram sehubungan masalah keterbatasan memori utama.