# OPERATING SYSTEM WEEK 10 ASSIGNMENT

# Given memory partitions of 100k,500k,200k,300k and 600k (in order), how would each of the First-fit, Best fit and worst –fit algorithms place processes of 212k,417k,112k and 426 k(in order)? which algorithm makes the most efficient use of the memory?

First-Fit:

212K is put in 500K partition. 417K is put in 600K partition.

112K is put in 288K partition (new partition 288K = 500K - 212K). 426K must wait.

Best-Fit:

212K is put in 300K partition. 417K is put in 500K partition. 112K is put in 200K partition. 426K is put in 600K partition. Worst-Fit:

212K is put in 600K partition. 417K is put in 500K partition. 112K is put in 388K partition. 426K must wait.

In this example, Best-Fit turns out to be the best

# Discuss in detail about

1. **Paging hardware - logical address to physical address mapping. (diagram)**

# TLB

1. **External fragmentation with solution**

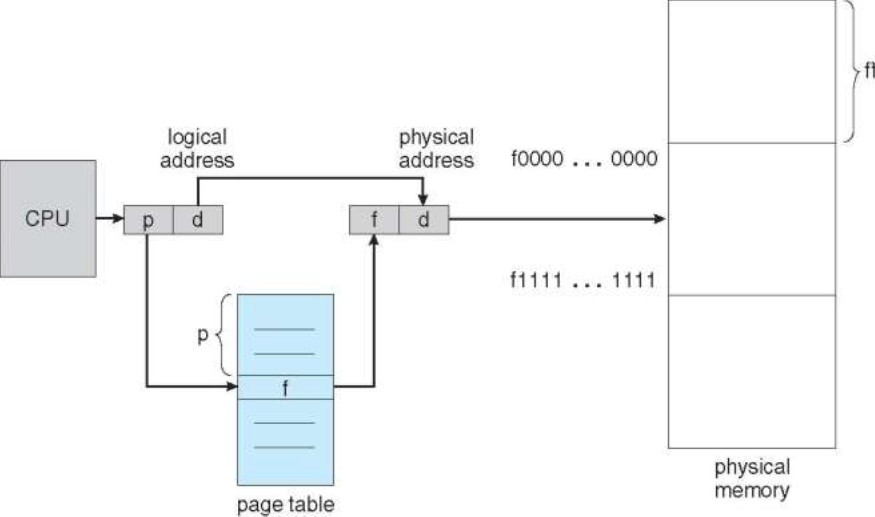
# Paging

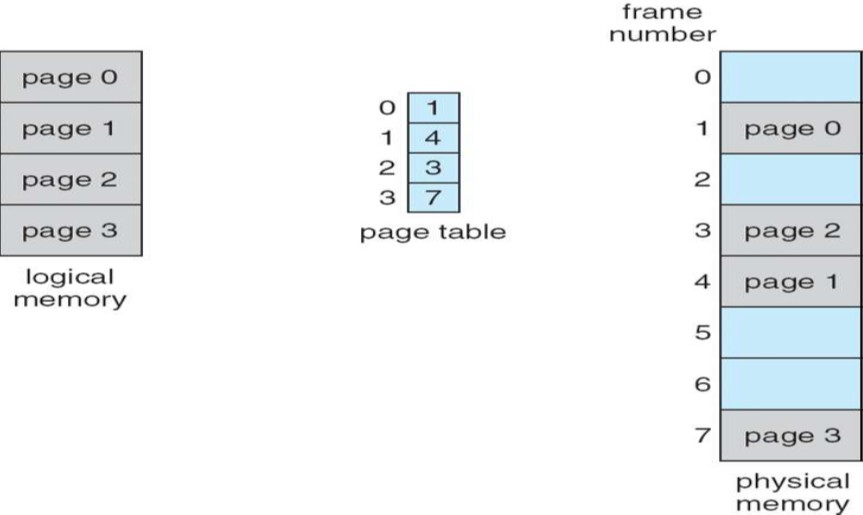
* Physical address space of a process can be non-contiguous; process is allocated physical memory whenever the latter is available
  + Avoids external fragmentation
  + Avoids problem of varying sized memory chunks Divide physical memory into fixed-sized blocks called frames
  + Size is power of 2, between 512 bytes and 16 Mbytes Divide logical memory into blocks of same size called pages\
  + Keep track of all free frames To run a program of size N pages, need to find N -free frames and load program
  + Set up a page table to translate logical to physical addresses
  + Backing store likewise split into pages
  + Still have Internal fragmentation

# 4.1 Address Translation Scheme

* Address generated by CPU is divided into:
  + Page number (p) – used as an index into a page table which contains base address of each page in physical memory
  + Page offset (d) – combined with base address to define the physical memory address that is sent to the memory unit

o For given logical address space 2m and page size2n





# logical address to physical address mapping

**Effective Access Time**

**·** Associative Lookup = time unit

o Can be < 10% of memory access time

* Hit ratio = α

o Hit ratio –percentage of times that a page number is found in the associative registers; ratio related to number of associative registers

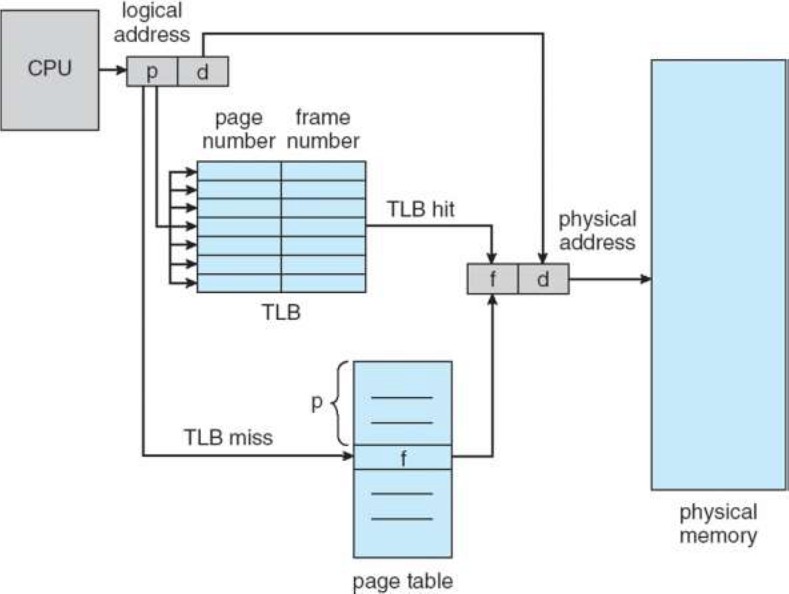
* Consider α = 80%, = 20ns for TLB search, 100ns for memory access
* Effective Access Time(EAT)

· EAT = (1 + ) α + (2 + )(1 – α ) = 2 + – α

* Consider α = 80%, = 20ns for TLB search, 100ns for memory access

o EAT = 0.80 x 100 + 0.20 x 200 = 120ns

* Consider more realistic hit ratio -> α = 99%, = 20ns for TLB search, 100ns for memory access EAT = 0.99 x 100 + 0.01 x 200 = 101ns



TLB

# EXTERNAL FRAGMENTATION

* + External fragmentation exists when there is enough total memory space to satisfy a request but available spaces are not contiguous.
  + First-fit and best-fit memory allocation suffers from external fragmentation.
  + Systems with variable-sized allocation units, such as the multiple partitions scheme and segmentation suffer from external fragmentation.