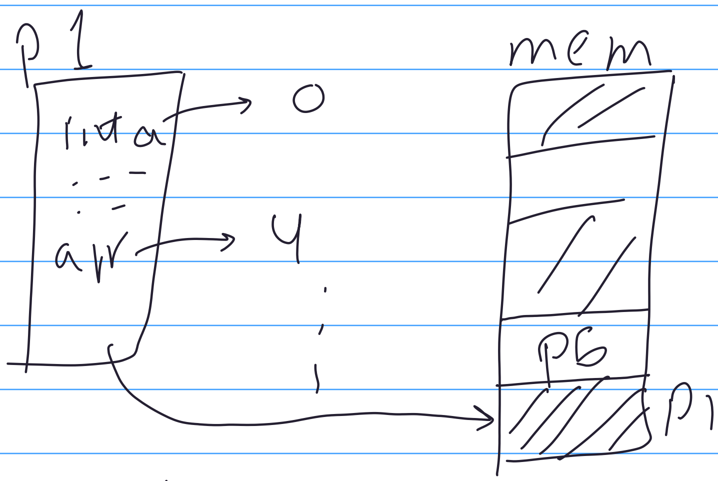
Main Memory



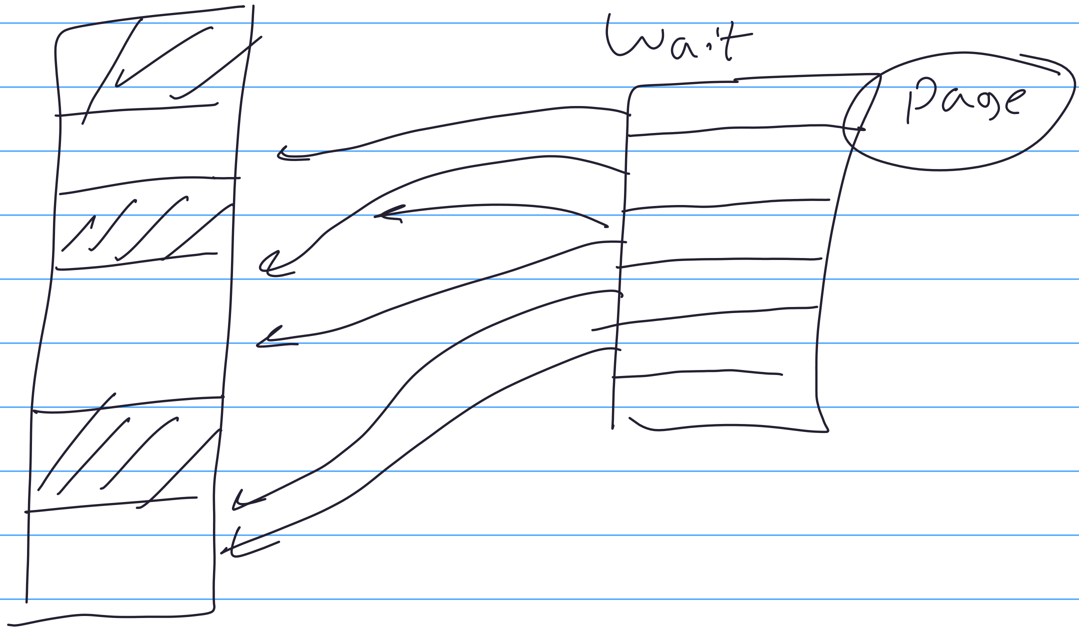
P2

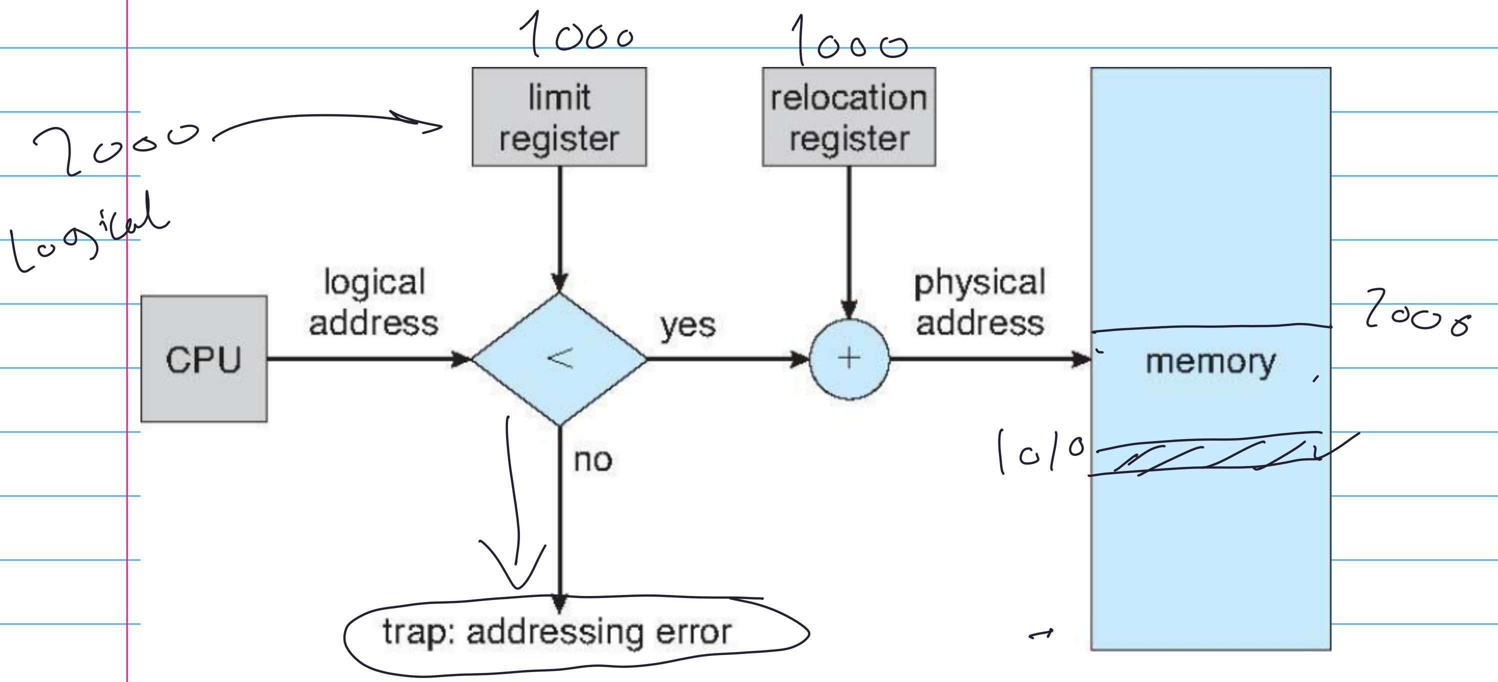
swap

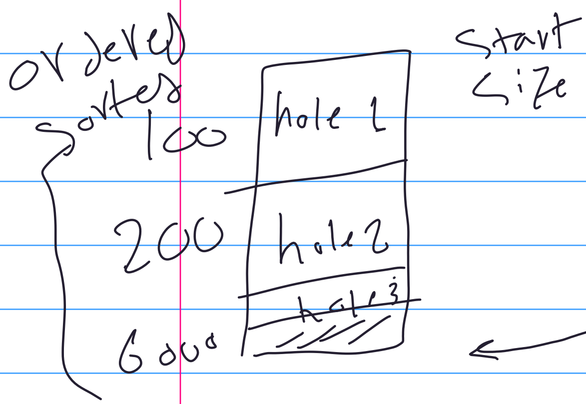
Compile time

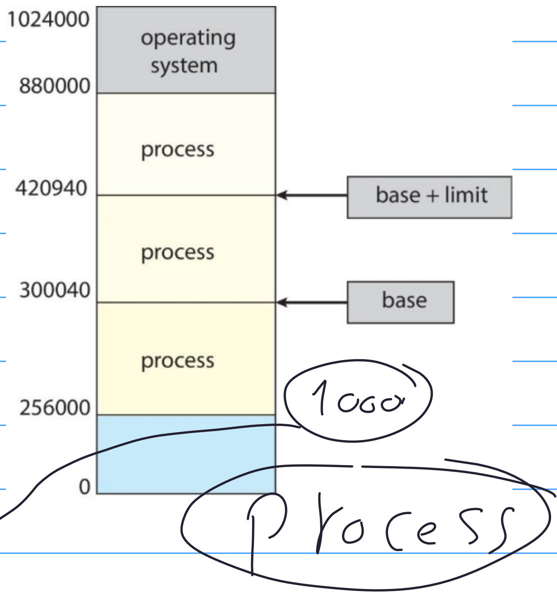
load time

execution time









- **First fit**. Allocate the first hole that is big enough. Searching can start either at the beginning of the set of holes or at the location where the previous first-fit search ended. We can stop searching as soon as we find a free hole that is large enough.
- **Best fit**. Allocate the smallest hole that is big enough. We must search the entire list, unless the list is ordered by size. This strategy produces the smallest leftover hole.
- **Worst fit**. Allocate the largest hole. Again, we must search the entire list, unless it is sorted by size. This strategy produces the largest leftover hole, which may be more useful than the smaller leftover hole from a best-fit approach.

CPU

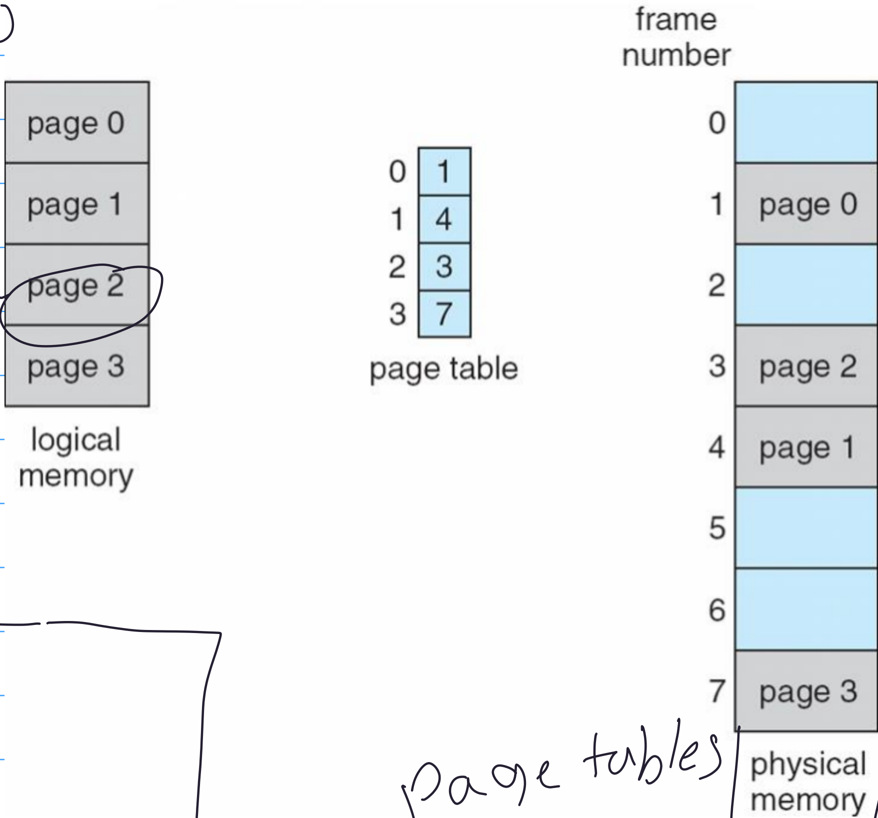
Page

Logical

memory

frame

Physical

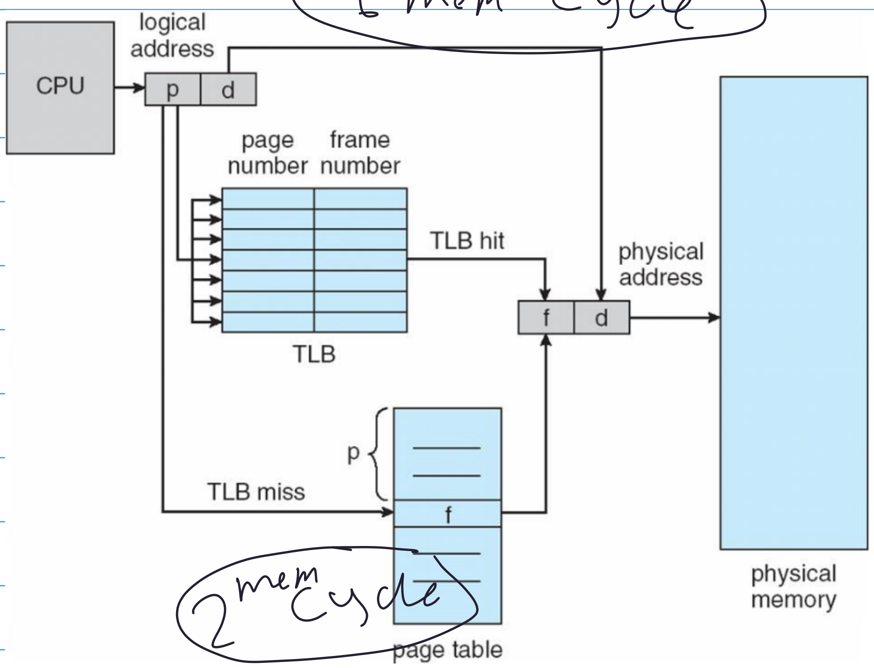


Cache

TLB

page tables

1 mem cycle



2 mem cycle

8.4 Consider a logical address space of 64 pages of 1,024 words each, mapped onto a physical memory of 32 frames.

a. How many bits are there in the logical address?

b. How many bits are there in the physical address?

$64 * 1024 \Rightarrow 32 * 1024$

$\boxed{\log_2(64 * 1024)}$ $\quad 2^5 \quad 2^{10}$

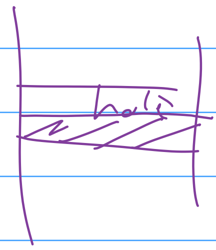
$2^6 \quad 2^{10} \quad 2^{16} \quad\quad 2^{15}$

16 bits $\quad\quad$ 15 bits

8.11 Given six memory partitions of 300 KB, 600 KB, 350 KB, 200 KB, 750 KB, and 125 KB (in order), how would the first-fit, best-fit, and worst-fit algorithms place processes of size 115 KB, 500 KB, 358 KB, 200 KB, and 375 KB (in order)? ~~Rank the algorithms in terms of how efficiently they use memory.~~

Process size	Goes to ...
115	300
500	600
358	750
200	350
375	392

Holes	Residual	
300	185	✓
600	100	✓
350	150	✓
200		✓
750	398	17 ✓
125	✓	



8.11 Given six memory partitions of 300 KB, 600 KB, 350 KB, 200 KB, 750 KB, and 125 KB (in order), how would the first-fit, best-fit, and worst-fit algorithms place processes of size 115 KB, 500 KB, 358 KB, 200 KB, and 375 KB (in order)? Rank the algorithms in terms of how efficiently they use memory.

Process size	Goes to ...
115	125
500	600
358	750
200	200
375	392

Holes	Residual
~~125~~	10
200	0
300	
350	
~~600~~	100
~~750~~	392

8.11 Given six memory partitions of 300 KB, 600 KB, 350 KB, 200 KB, 750 KB, and 125 KB (in order), how would the first-fit, best-fit, and worst-fit algorithms place processes of size 115 KB, 500 KB, 358 KB, 200 KB, and 375 KB (in order)? Rank the algorithms in terms of how efficiently they use memory.

Process size	Goes to ...
115	750
500	635
358	600
200	350
375	wait

Holes	Residual	
125		
200		
300		
350	150	
~~600~~	242	
~~750~~	~~635~~	135

125
135
150
242
300

8.20 Assuming a 1-KB page size, what are the page numbers and offsets for the following address references (provided as decimal numbers):

1024

a. 3085

b. 42095

c. 215201

d. 650000

e. 2000001

page number = int(address / page size)

634 , 784

page number = address // page size
offset = address % page size

Page number	offset
3	13

8.23 Consider a logical address space of 256 pages with a 4-KB page size, mapped onto a physical memory of 64 frames.

a. How many bits are required in the logical address?

b. How many bits are required in the physical address?

$\log_2 (256 * (4 * 1024)) = \log_2 (2^8 \cdot 2^2 \cdot 2^{10}) = 20 \text{ bit}$

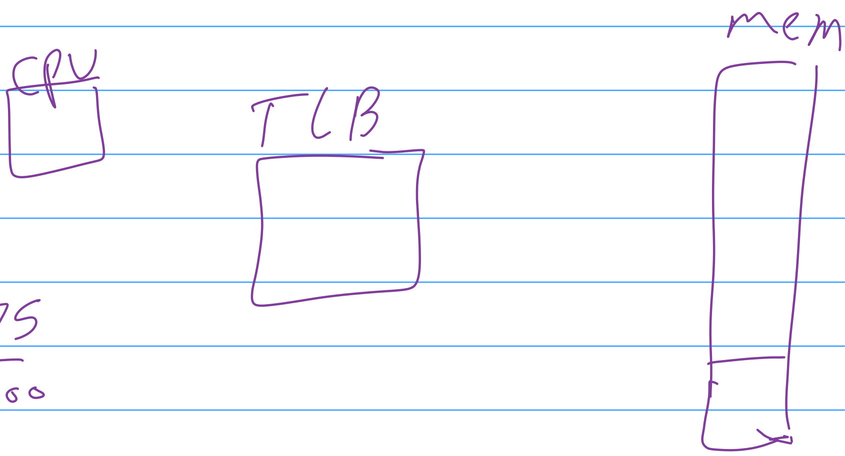
$2^6 \cdot 2^{12} = 2^{18}$

8.25 Consider a paging system with the page table stored in memory.

a. If a memory reference takes 50 nanoseconds, how long does a paged memory reference take?

b. If we add TLBs, and 75 percent of all page-table references are found in the TLBs, what is the effective memory reference time? (Assume that finding a page-table entry in the TLBs takes 2 nanoseconds, if the entry is present.)

$\boxed{a}$ $50 * 2 = 100$ nsec



$\boxed{b}$

$$(0.2 + 50) * \frac{75}{100} + (0.2 + 50 + 50) * \frac{25}{100} = 64.5 \text{ nsec}$$

8.28 Consider the following segment table:

Segment	Base	Length
0	219	600
1	2300	14
2	90	100
3	1327	580
4	1952	96

What are the physical addresses for the following logical addresses?

a. 0,430 <600 430+219 = 649

b. 1,10 2300+10 = 2310

c. 2,500 trap

d. 3,400 1327+400 = 1727

e. 4,112 trap

		Length
	430	600
	10	14
trap	500	100
	400	580
trap	112	96