I how is it used?

content borrowed from
www.cs.columbia.edu/~junfeng/10sp-w4118/lectures/...

11/13/16 ①

paging

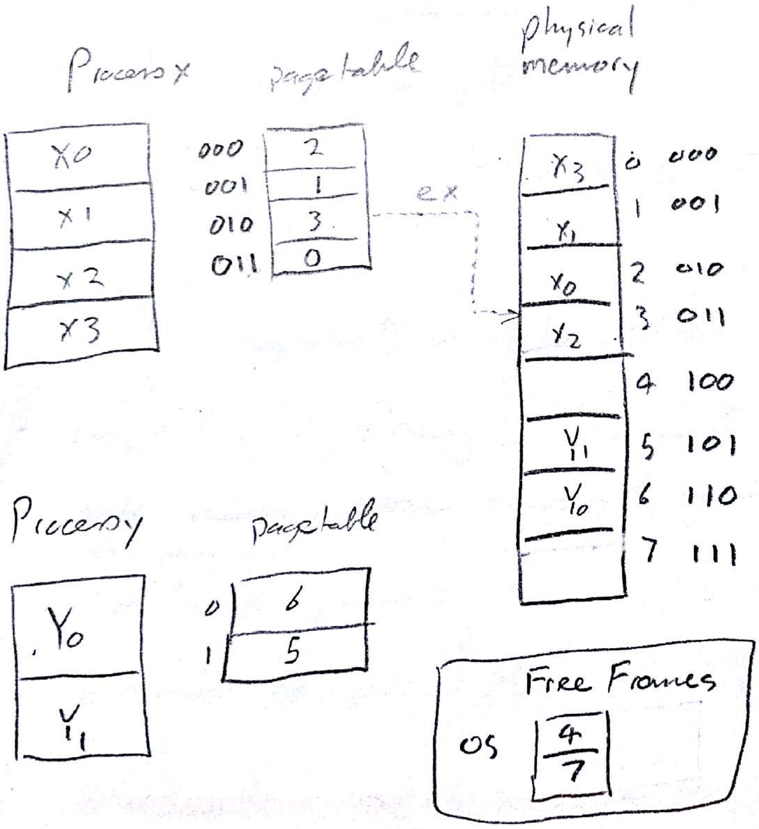
frames (= length)

16 bits 6 for frame# then have 2^6 frames each with 2^{10} words

process page = frame size

slide 37 – 41

each page table kept with process control block



say have 8 bit address space

have 8 frames (3bits)

frames have $2^5 = 32$ words

lle

~~but still need all of process in mem at once!~~
~~also internal frag on last block of process.~~

ex 8 bit logical address for process X

|— logical —|

$$\underbrace{010}_{page\#}\ \underbrace{00001}_{offset}$$

+ limit (if used?)

11/13/16 ②

Segmentation ~~like paging with diff size frames~~

can have diff sized segments (~~like frames in paging~~)

like paging logical address has 2 parts

$$\overbrace{\underbrace{010}_{\text{seg\#}}\quad\underbrace{00001}_{\text{offset}}}^{\text{logical}}$$

diff is the segment table

has base & length

↑ like paging

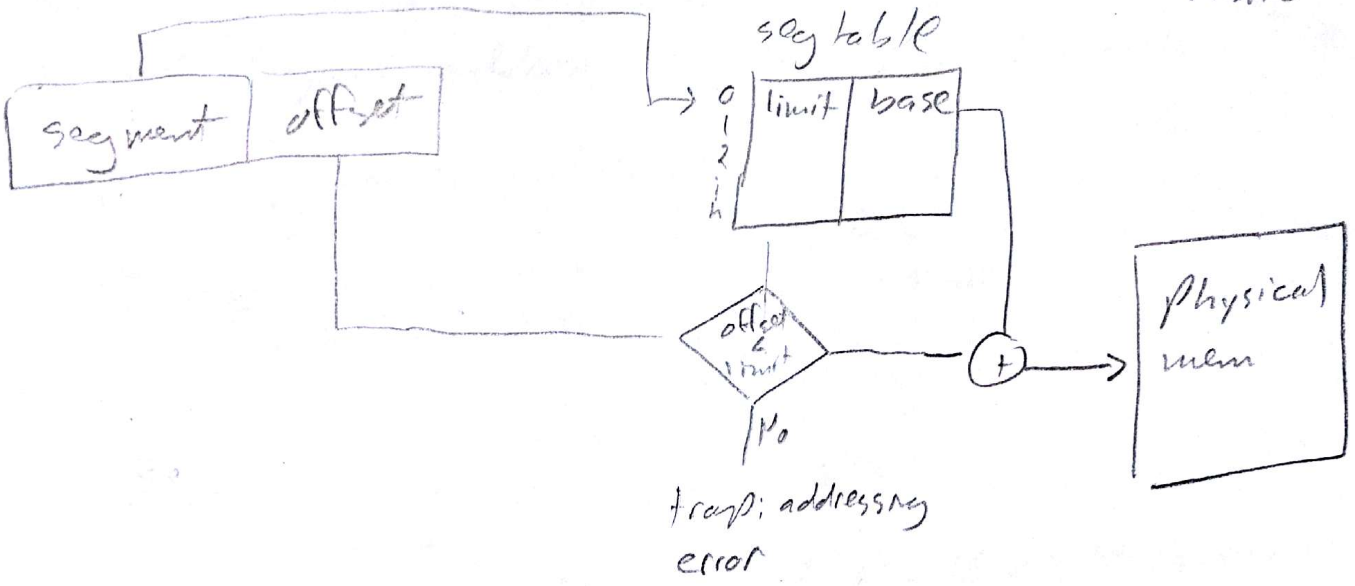
↑ max size

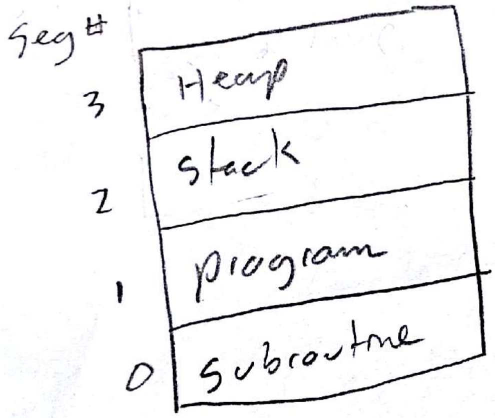
- these segments map to blocks of memory
- no internal frag. (external though) best fit? first fit?
- ~~programmers~~ compilers can control how process is segmented (data, program)

bit more complex than simple paging.

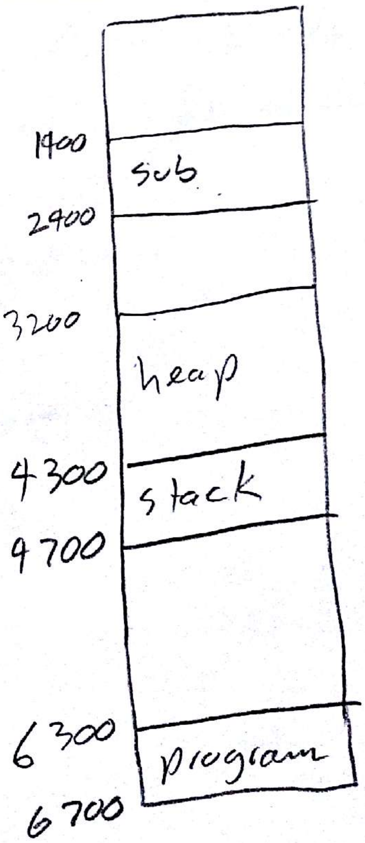
still all process in mem at once

11/13/16





	limit	base
0	1000	1400
1	400	6300
2	400	4300
3	1100	3200



also bits to determine read, write, dirty stopped here 11/19/16 first class [illegible]

paging

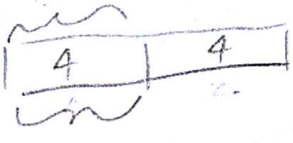
11/13/16 ④

6 bit logical address   8 bit physical address

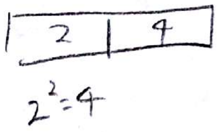
each frame is 16 bytes

① how many page table entries

② what if have 8 bit logical address?



$2^4 = 16$



③ how many physical pages? ($2^4 = 16$ in both cases)

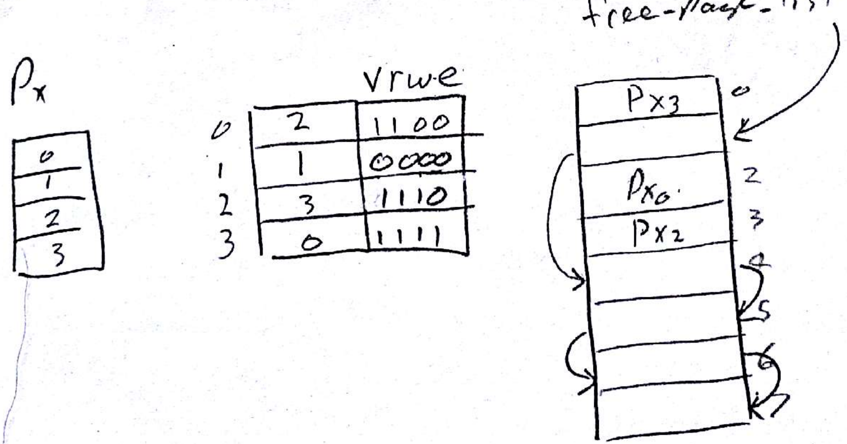
④ how many logical (6 bits) $2^2 = 4$ (8 bits) $2^4 = 16$

add some bits to page table

valid bit: map to a valid physical page

read/write/execute bits

all checked by MMU on each mem access



		vrwe
0	2	1100
1	1	0000
2	3	1110
3	0	1111

can allocate from free-page-list. one page at a time from head of this list.