Simple Paging (Real Paging)

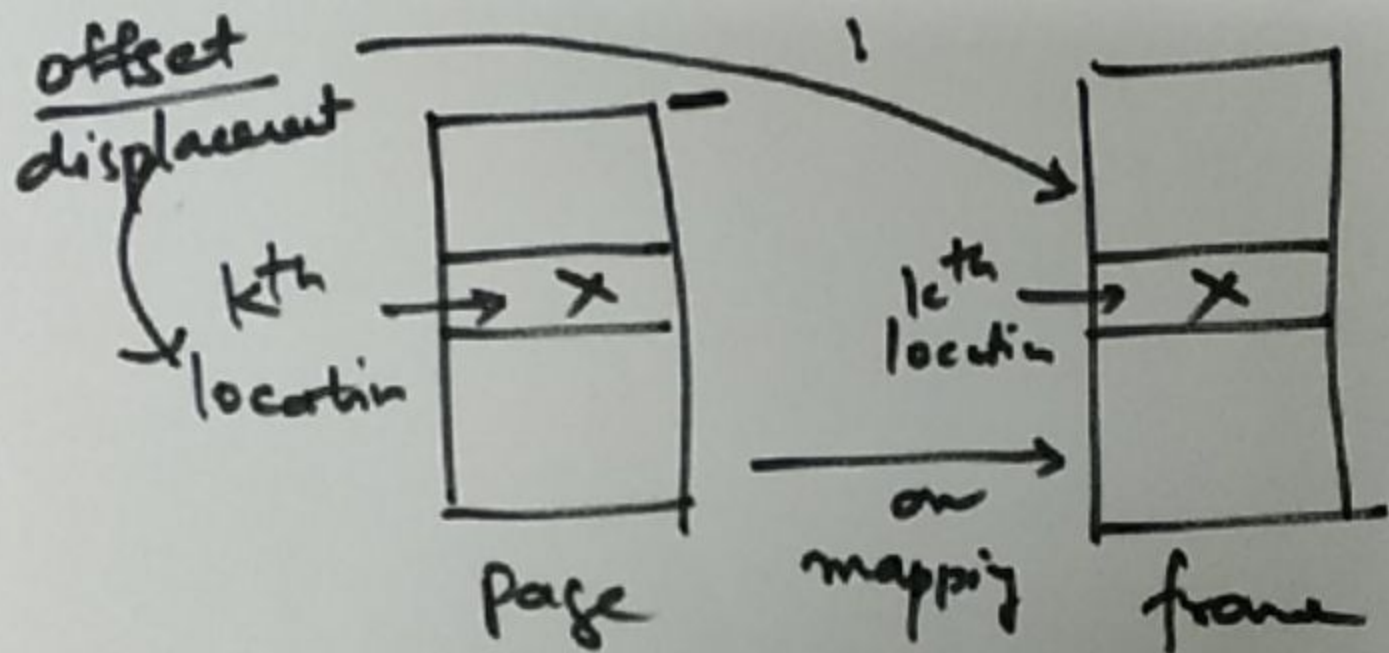— Mapping: Page → Frame

$$\underline{\underline{PT}} \text{ (page table)}$$

— Page and frame both of same size

Page (2k) → Frame (2k)

" (4k) → " (4k)

⋮

offset
displacement

k^{th} location → page (X)

lc^{th} location → frame (X)

on mapping

Page → frame

Address Translation

p#	d

LA

Page No. — offset / displacement in page p

p → f

page table

f#	d

PA

LA

p	d

f	d

PA → Mem loc.

p → | | |
| f |
| | |

page table.
(main memory)
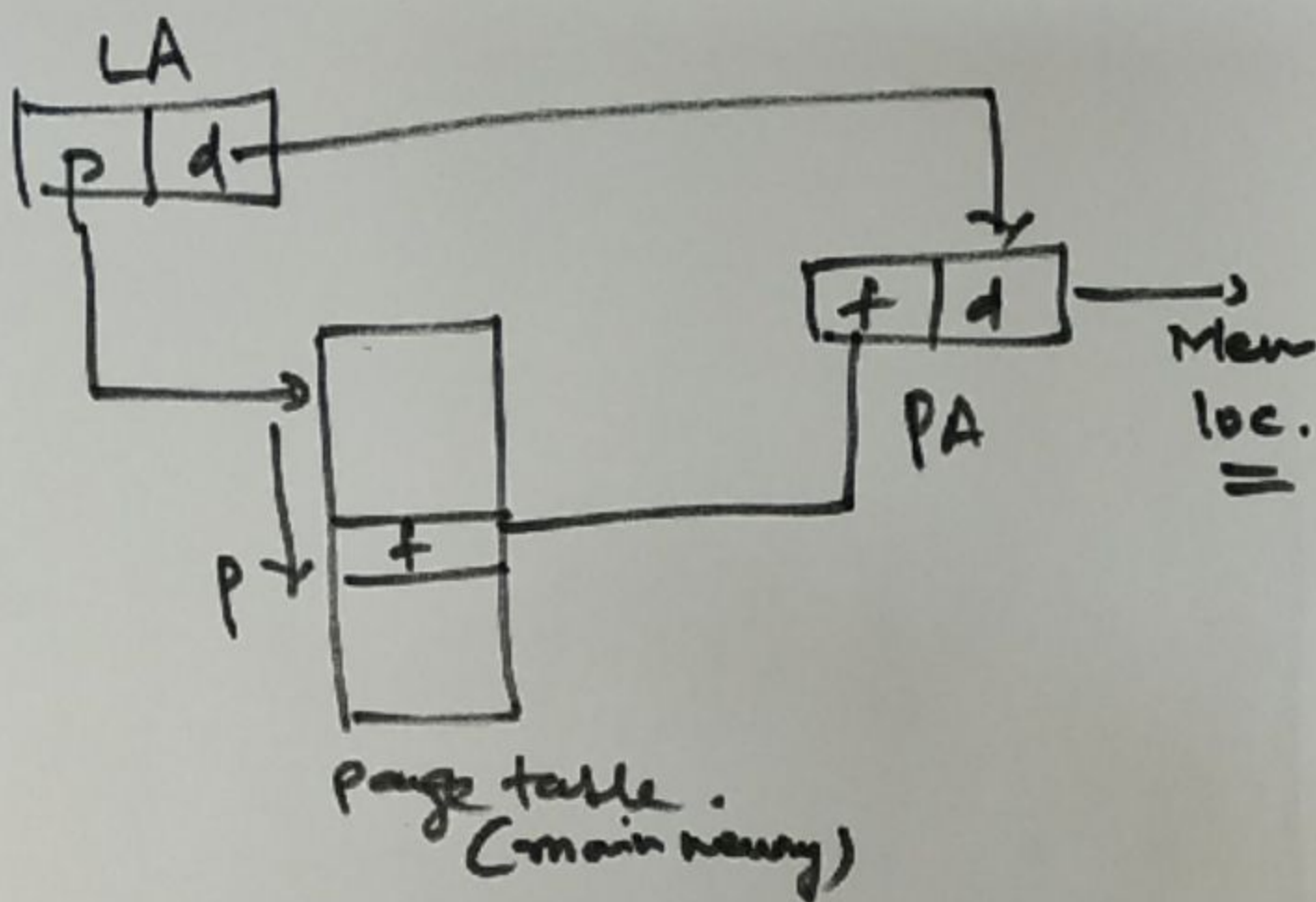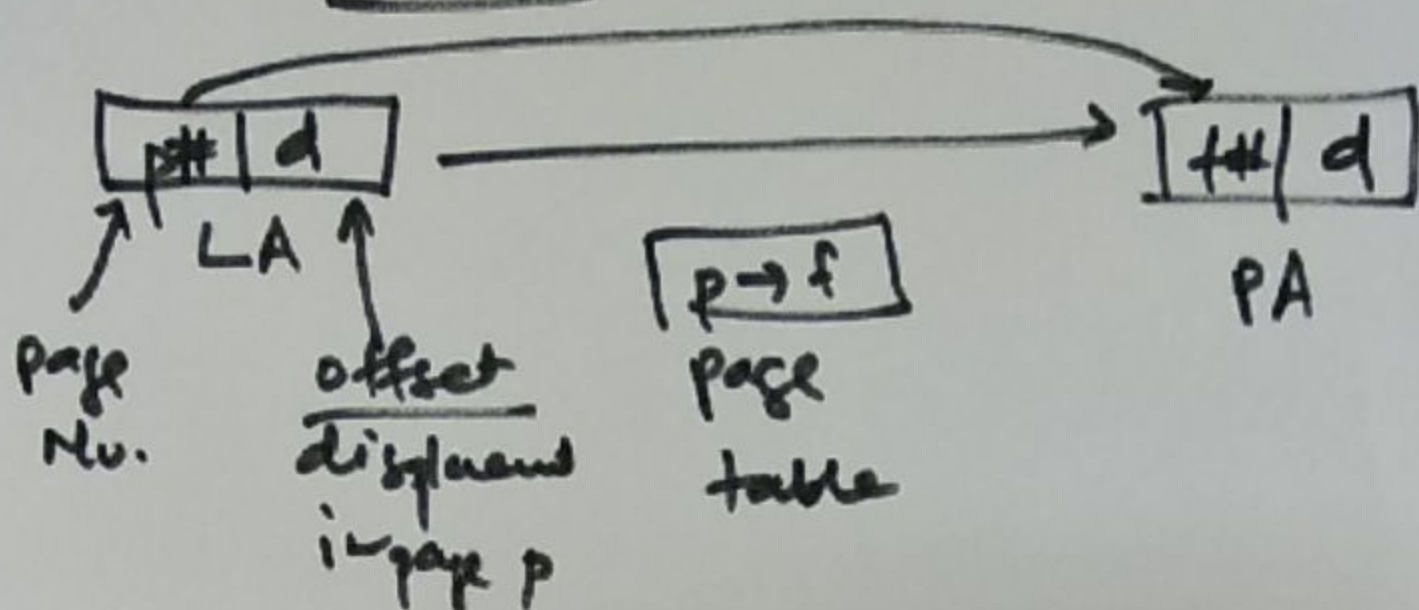
EAT
Effective Access time

$$\sim m_a + m_a = 2m_a$$

page ⟨ data
       ⟩ stack
       ⟩ instruction

③
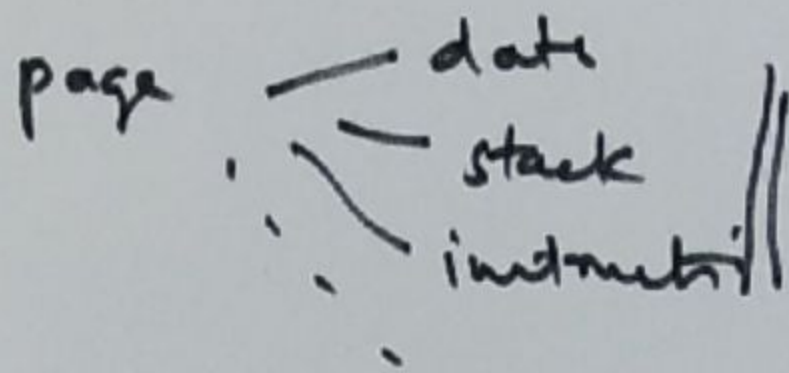
Segmentation → User/Programmer's
                         View

↳ Code Segment — 16K
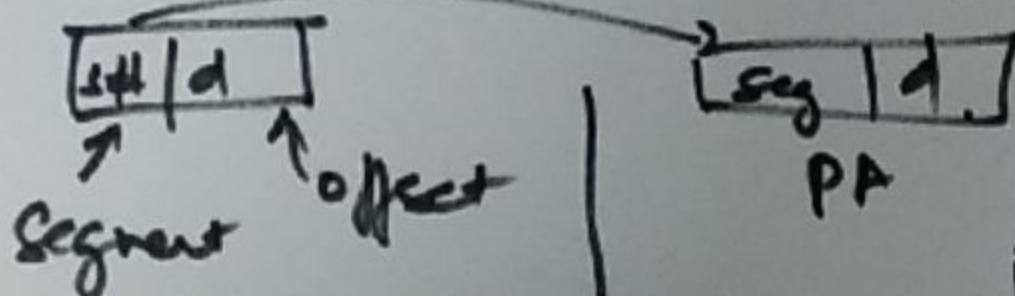 Data Segment — 2k
  Stack Segment === 1k

— All segments are variable is size

— LA → PA   EAT = m_a
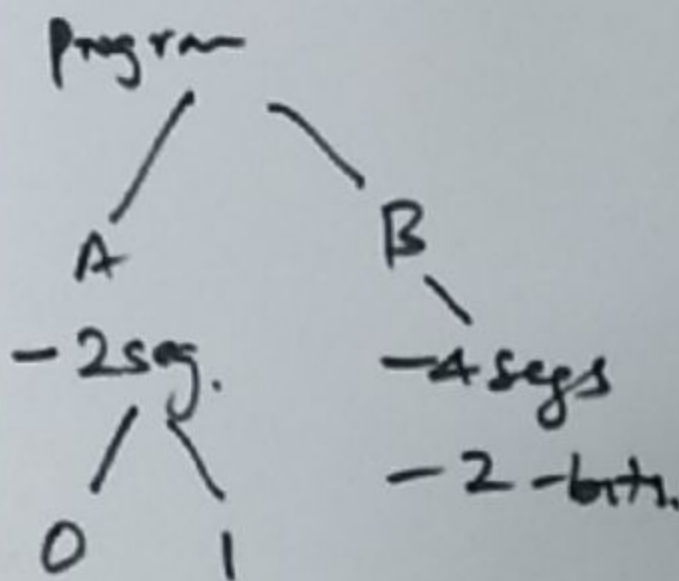   + m_o
                                = $2m_a$

| s# | d | | seg | d |
  ↗   ↑offset   PA
Segment

Segment table

s#	size

— size of segment — important parameter

offset

Address

No. of segments

segment
↓
— length
↓
8 bytes   64 byte
| ↓
36 bits   more
(offset)   bits.

Program

A   B
—2seg.   —4 segs   — Good
/↙↘   ↘   (User)
0   1   —2-bit.   — Complex

(5)
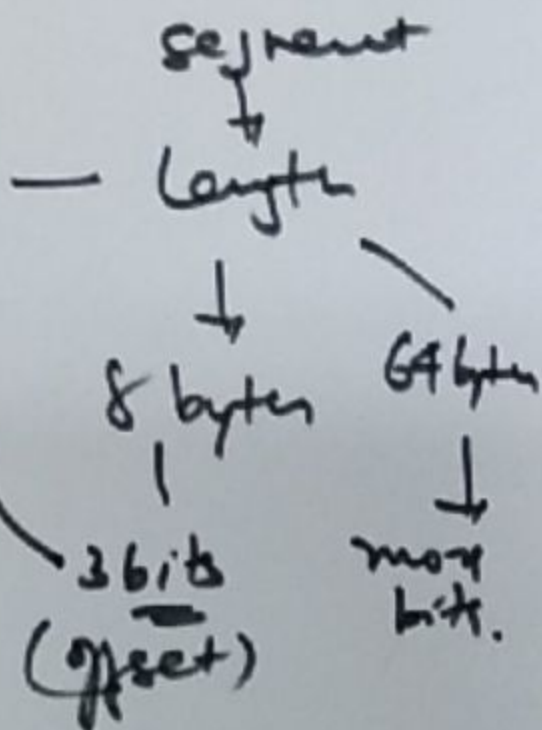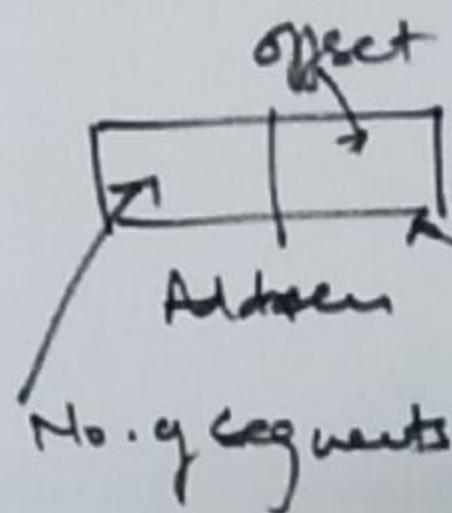
- (Real) Pagy (Simple Pagy)
- (Real) Segmentation (Simple Sequential)

    — Non-Contiguous

   → All pages ⎤
     — All segments ⎦ in memory

Program A	Program B
4 pages	7 pages

Simple seg'gs — $\dfrac{4 \text{ frms}}{3 \text{ frms}}$ 7 frms

       (A wont)

Simple Segmentation — 2 segments ↓ memory

A complete program or program is needed in memory
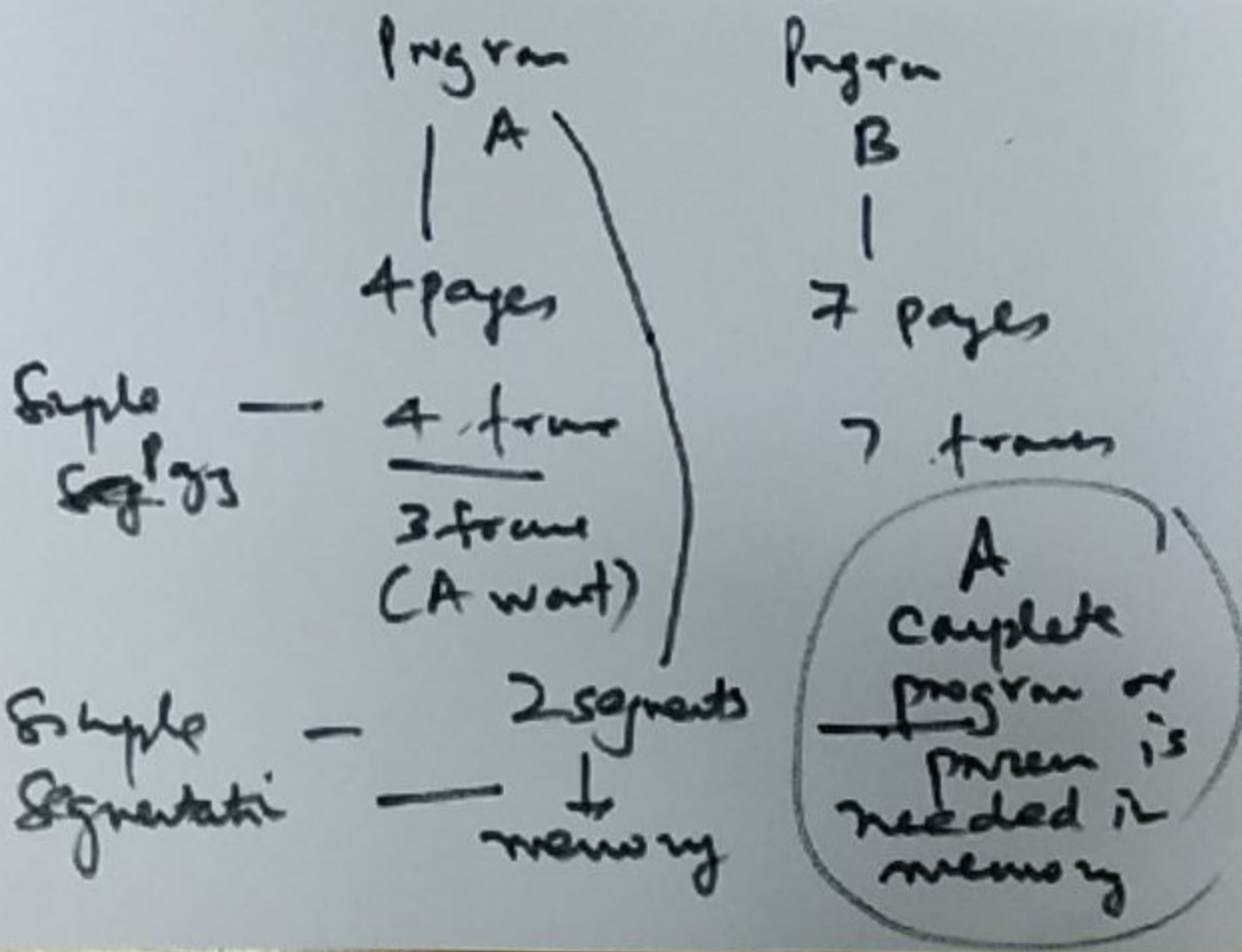
Page / Segment Table Implementation (6)

(1) Hardware — Page table
(faster) — Segment table
└ Limited size → limited program space

(2) Page table
— Memory — Specific registers

Page PTBR   PTLR
    Page Table   Page table
    Base Reg.   length reg.

Seg. STBR   STLR
   Segment   Segment
   table Base   table length
   reg.   register
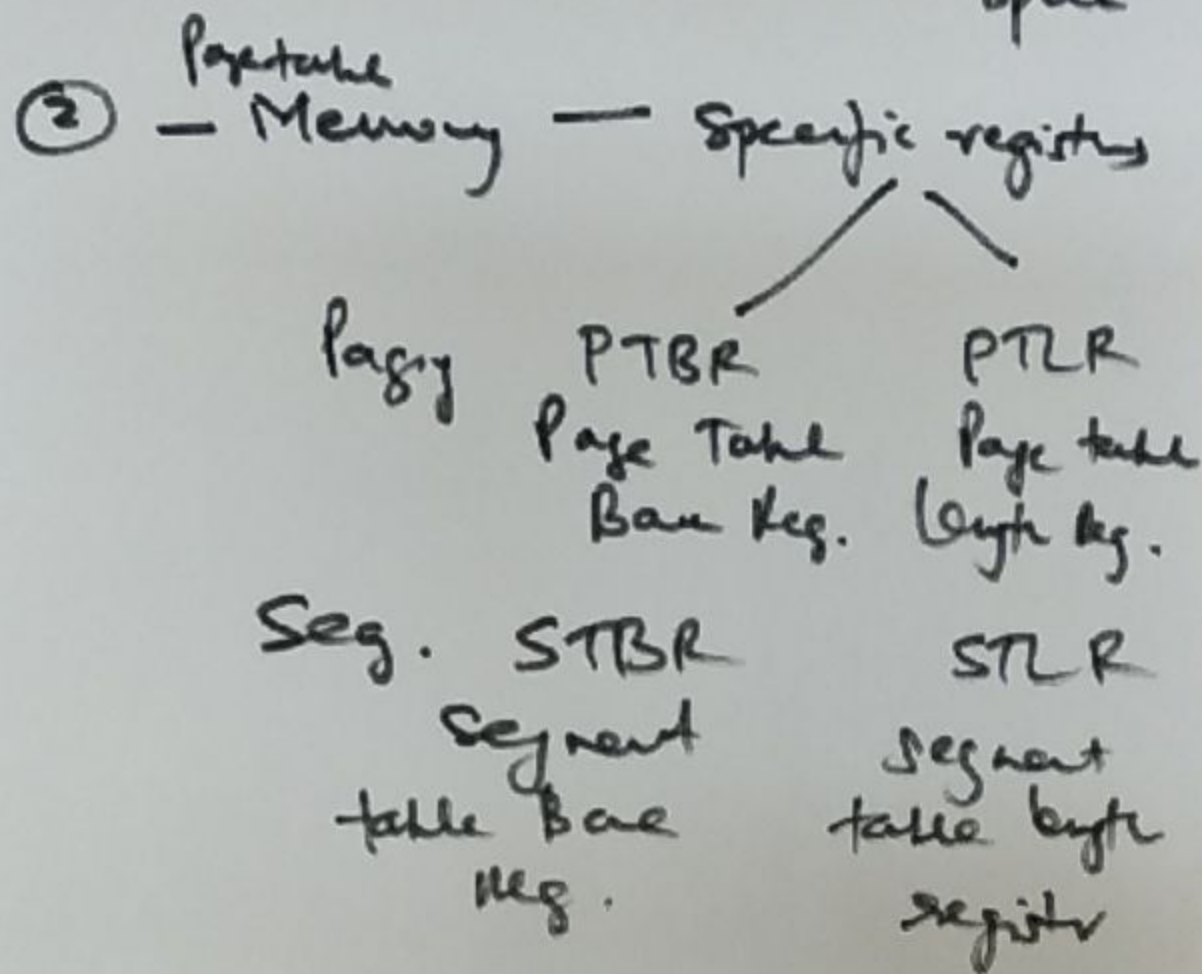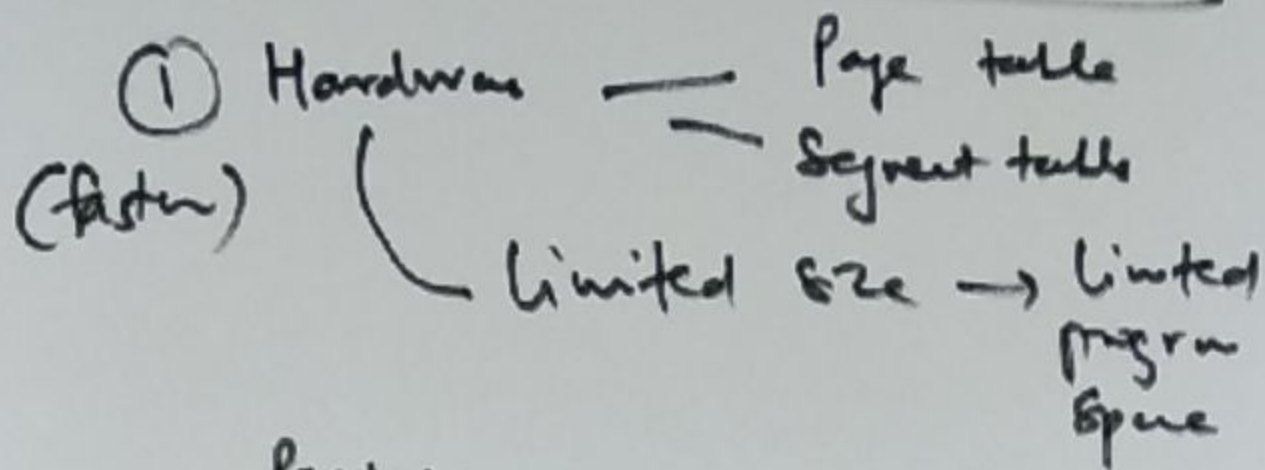
Program A ———switch———→ Program B

⑦

PBR ——→ Pg0 | a
   ↓   1
   ↓   2
PTLR ——
   b

PTBR ——→ Pg 0 | c
              1
MLR→
   d

o/s

EA → ti
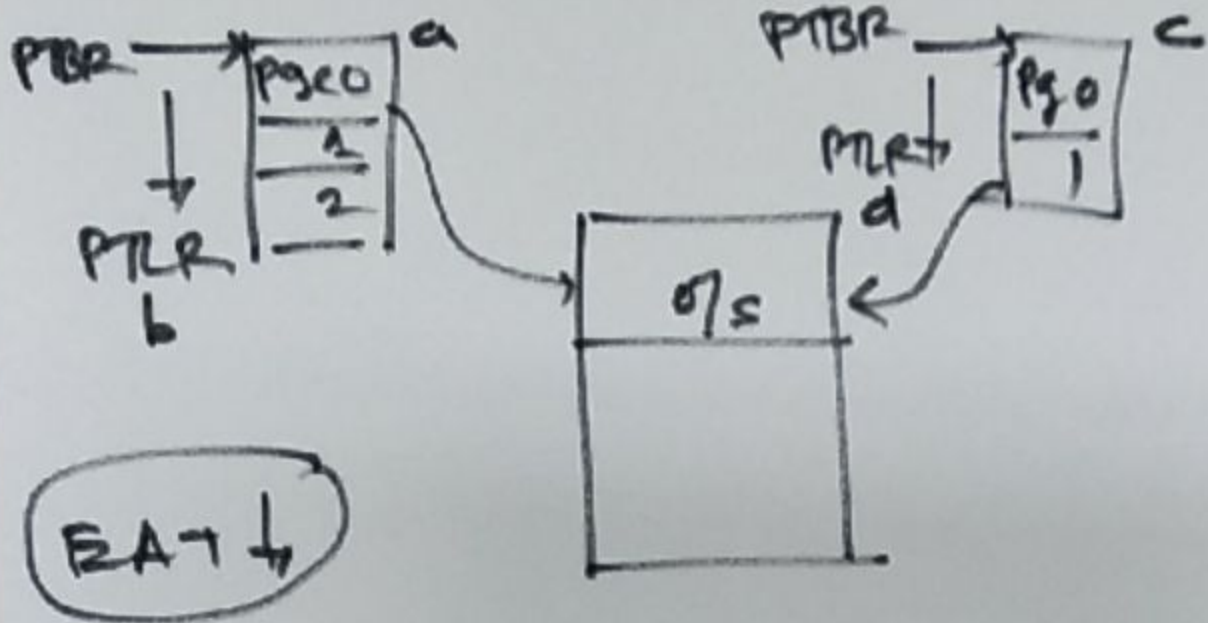
| ✗ c |
PTBR

| ✗ d |
PTLR

— STBR, STLR

STBR   STLR

segment table A   segment table B

③ Associative Memory (limited) ⑧
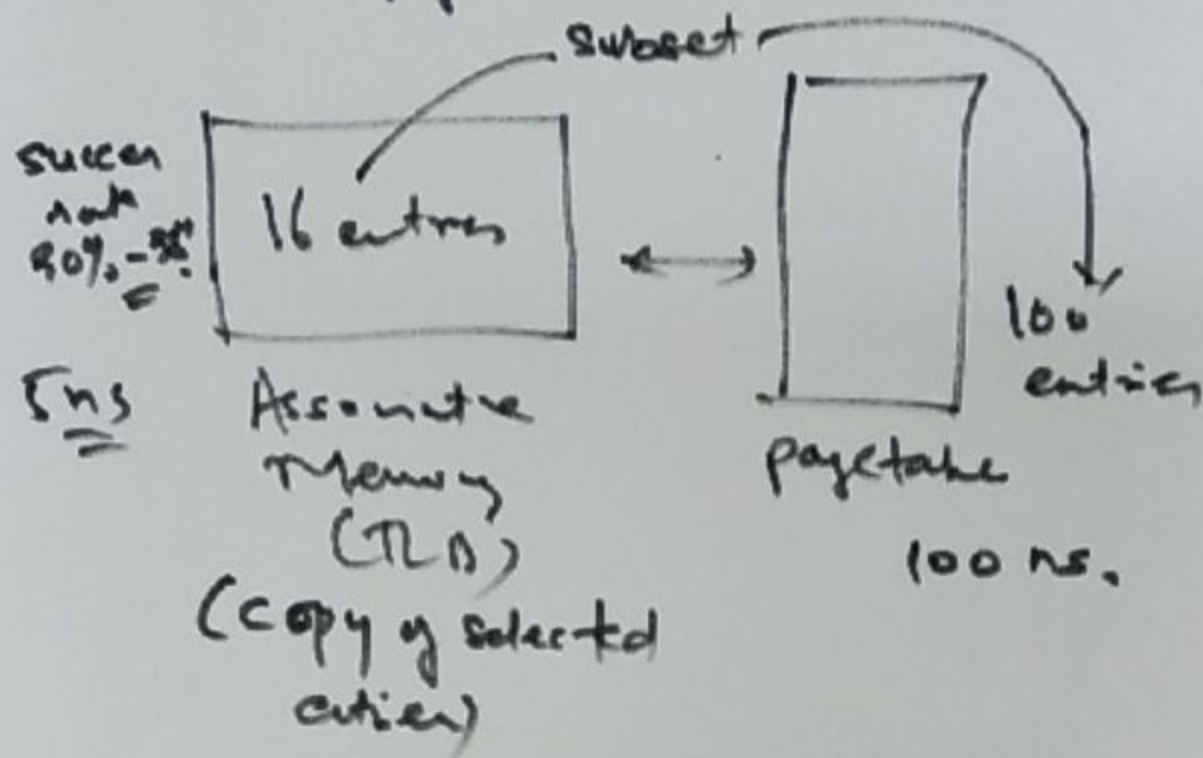(TLB (Translation Lookaside buffer)

↳ Store page table

succen
not
80%-98%

16 entries

5ns   Associative
        Memory
        (TLB)
        (copy of selected
         entries)

subset

100
entries

Pagetable

100 ns.

For LA→PA ⟶ Looks up TLB
    mapping

                    TLB   TLB
                    Hit   Miss
              probability (90%)   (10%)
              of success
              of getting into   (1- TLB Hit)
                 in TLB

(pag + TLB)   ⑤

LA

| p | d |

TLO Hit (A) →| f | d |

5ns

TLB Hit

$x\%$, t_m TLB
                               Miss
                               (B)

Updated

mem
loc.
(ma)
(100ns)
(105ns)

Path = 5+100
A   = 105ns

| f |

Page table
(Memory)

Path = 5+100ns
D   + 100ns

      = 205 ns

```
┌─────────────┐
│   page 0    │
│        1    │
│        2    │
│        3    │
│        4    │
└─────────────┘
program A
```

```
      ┌───┬───┐          ┌──────────┐
   0  │   │ 0 │ ───────→ │ frame 0  │
      ├───┼───┤          ├──────────┤
   1  │   │ 1 │          │ from 1   │
      ├───┼───┤          └──────────┘
   2  │   │ X │ ─┐
      ├───┼───┤  │
   3  │   │ X │  ├─→ page not
      ├───┼───┤  │      in memory
   4  │   │ X │ ─┘
      └───┴───┘
```

— Runny a program in much
   lesser space — Virtual Memy