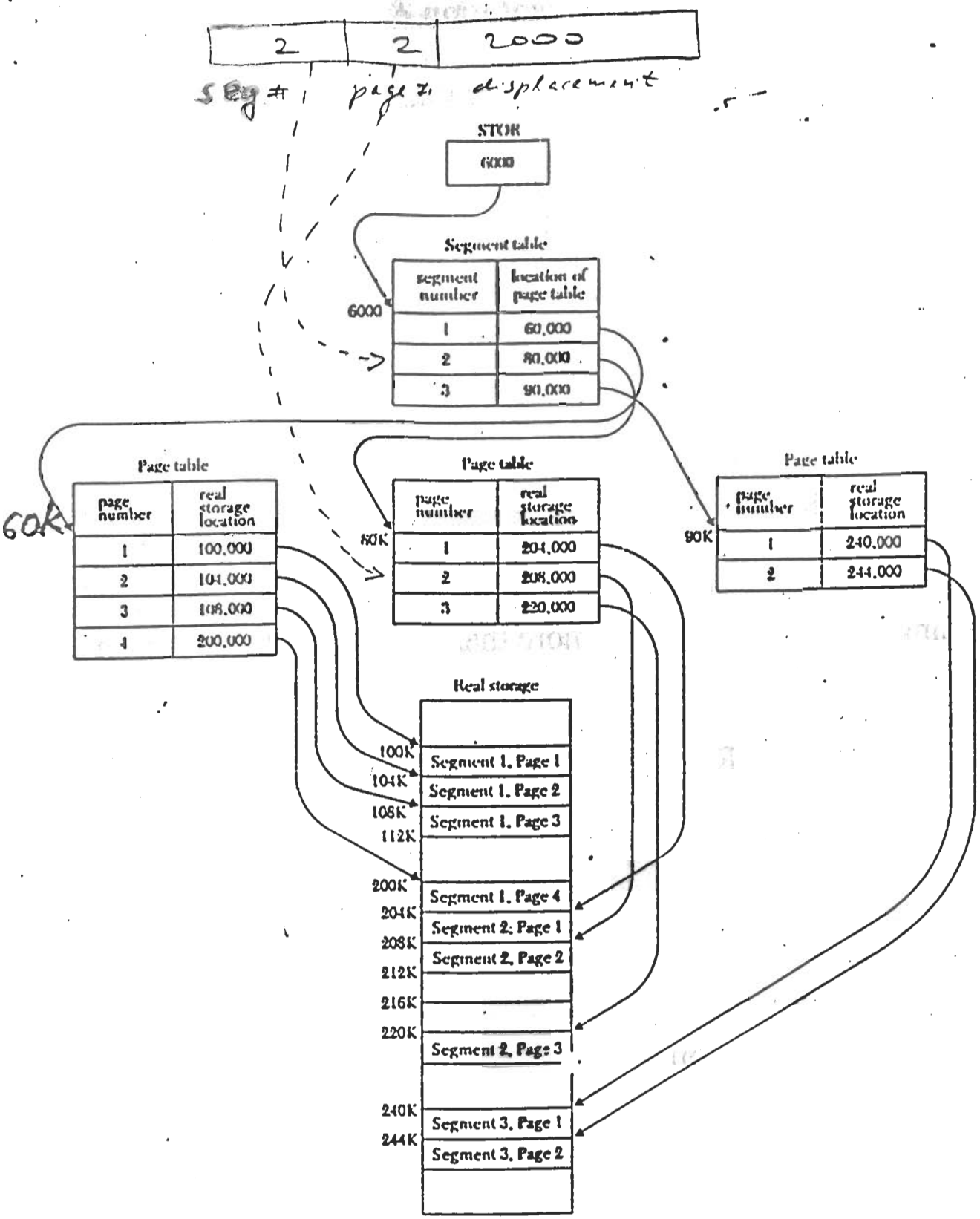
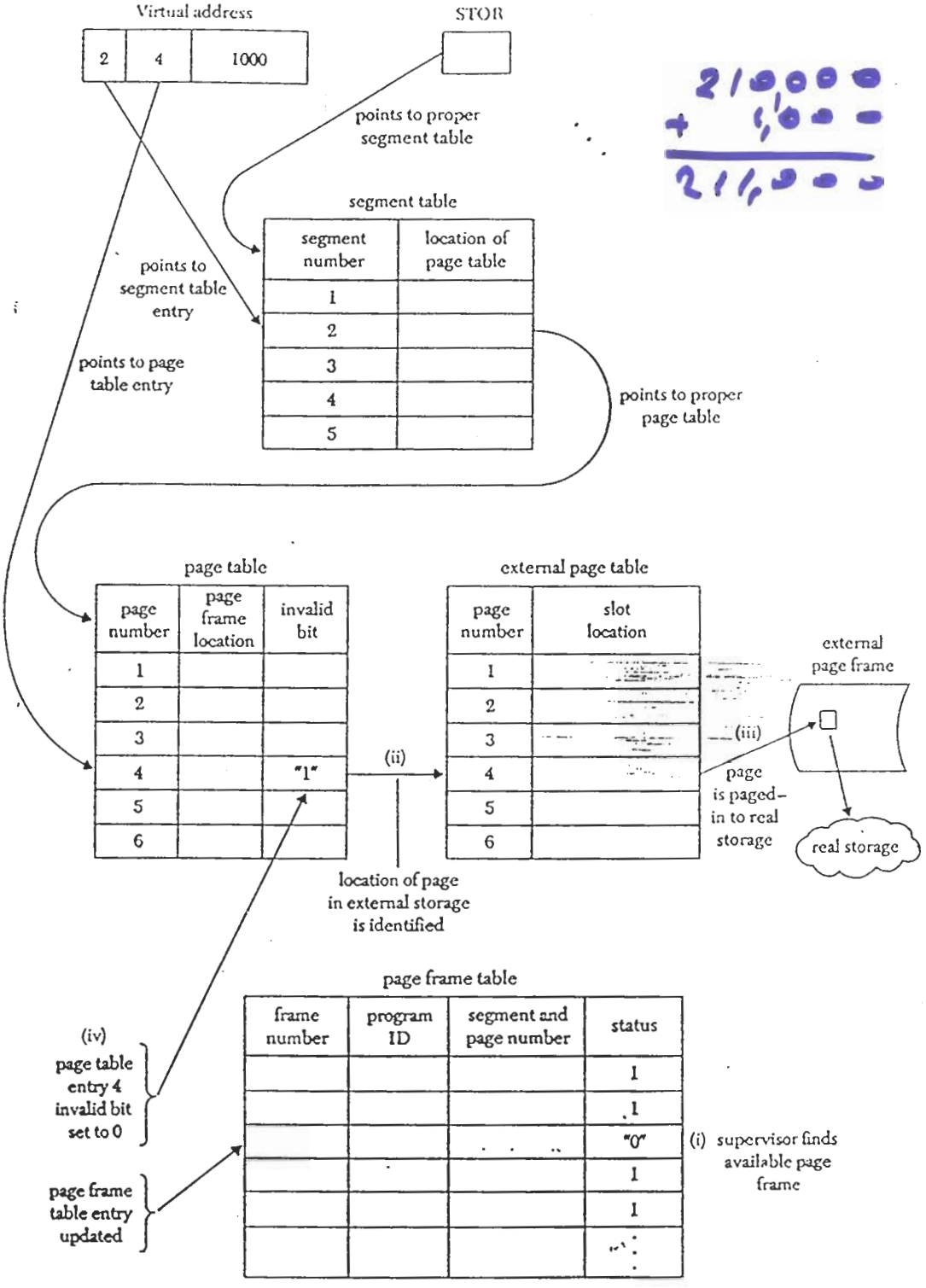

2
2
2000
seg #
page #
displacement
STOR
6000
Segment table
segment number
location of page table
6000
1
60,000
2
80,000
3
90,000
60K
Page table
page number
real storage location
1
100,000
2
104,000
3
108,000
4
200,000
80K
Page table
page number
real storage location
1
204,000
2
208,000
3
220,000
90K
Page table
page number
real storage location
1
240,000
2
244,000
Real storage
100K
Segment 1, Page 1
104K
Segment 1, Page 2
108K
Segment 1, Page 3
112K
200K
Segment 1, Page 4
204K
Segment 2, Page 1
208K
Segment 2, Page 2
212K
216K
220K
Segment 2, Page 3
240K
Segment 3, Page 1
244K
Segment 3, Page 2



Interpreting and Referencing a Virtual Address

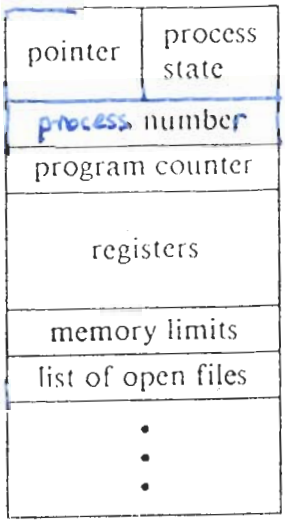


Figure 4.2 Process control block.

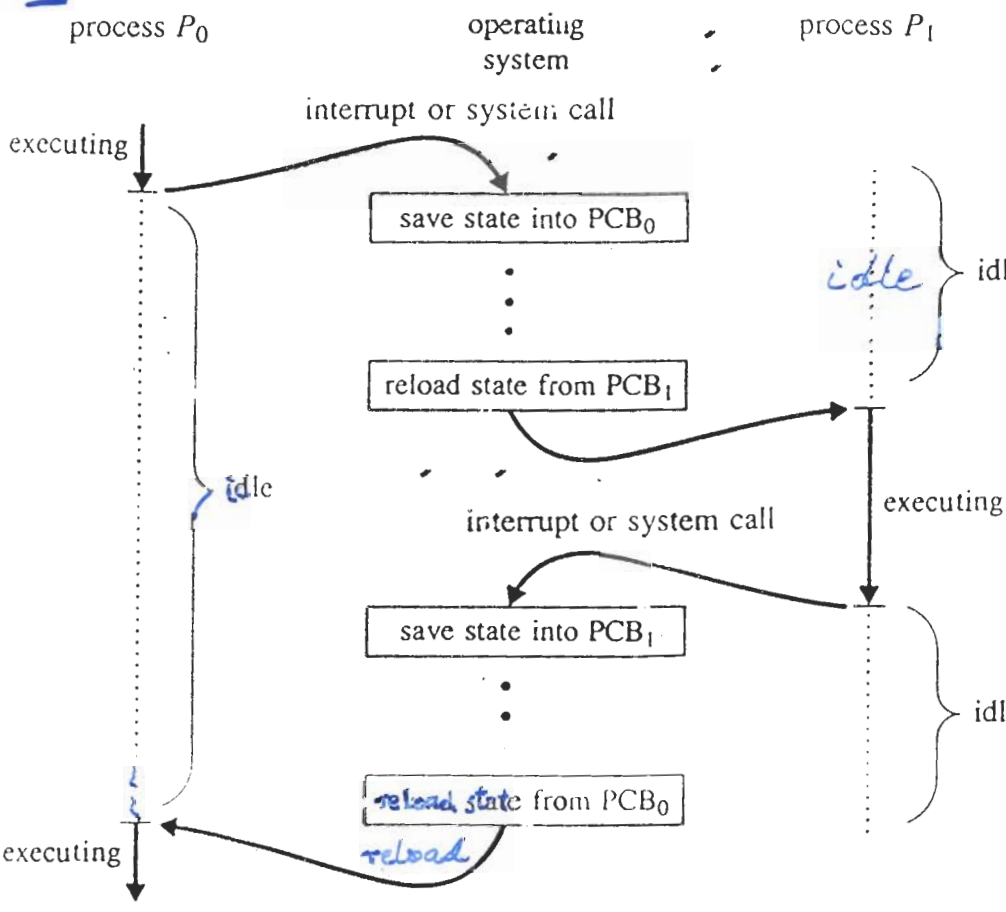
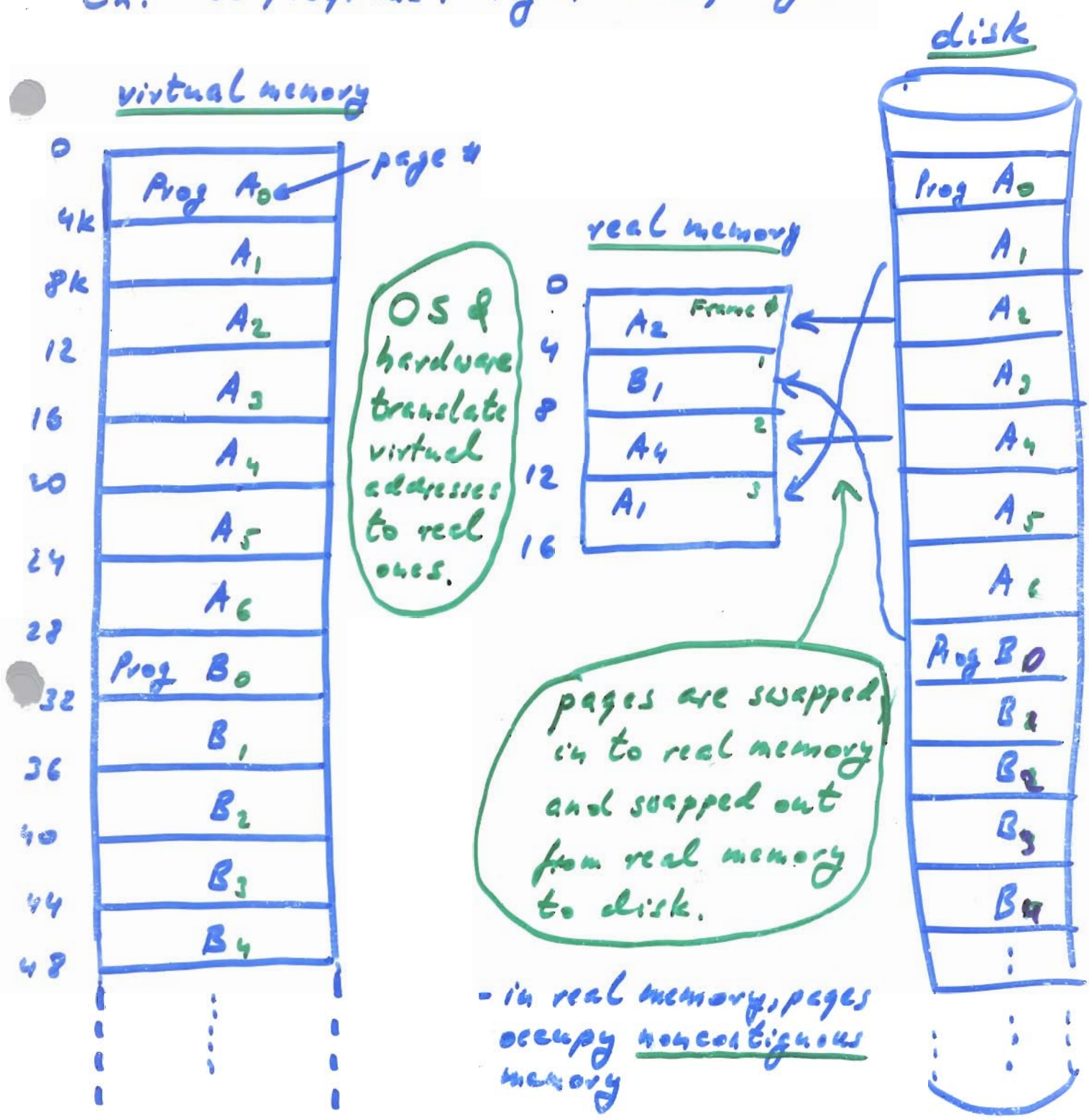


Figure 4.3 The CPU can be switched from process to process.

Ex. Two programs: Prog A = 25kB, Prog B = 20kB



- in real memory, pages occupy noncontiguous memory

- programs are assigned contiguous virtual addresses
- virtual memory is much larger than real memory

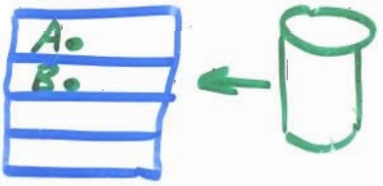
- programs' pages are physically stored on disk or in real memo

How pages are swapped in and swapped out?

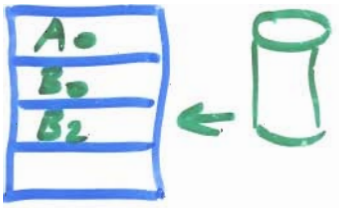
1. Initially real memory is empty

2. A_0 is loaded and executes, after a while A_0 requests I/O

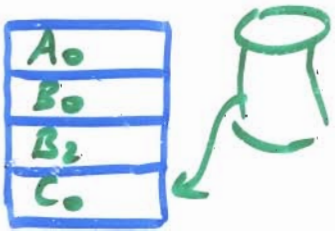
3. B_0 is loaded and starts to execute



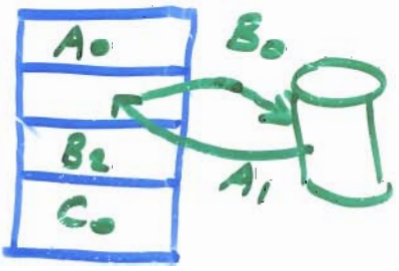
4. B_0 calls page B_2, after a while B_2 requests I/O



5. C_0 is loaded and runs, after a while C_0 requests I/O



6. A_0 gets data and resumes running. After a while, it calls A_1.



A_1 is <u>not</u> in real memory. It needs to be brought to real memory. There is no room in real memory. B_0 may be swapped out (LRU page - Project #2). A_1 swapped in and replaces B_0. WHEN B_0 is CALLED NEXT TIME, IT CAN BE LOADED INTO ANOTHER PAGE.