The Clock Algorithm

Problem Solving Lab for CS111

Stanford CS111ACE, Spring 2026

Today

- The Clock Algorithm
- Q&A + General Assignment 5 work time!

Demand Paging

Think-Pair-Share (3-min)

- How many possible virtual pages are there in our scheme?

x86-64 with 4KB pages has 36-bit virtual page numbers and 40-bit physical page numbers.

- How big is the page map?

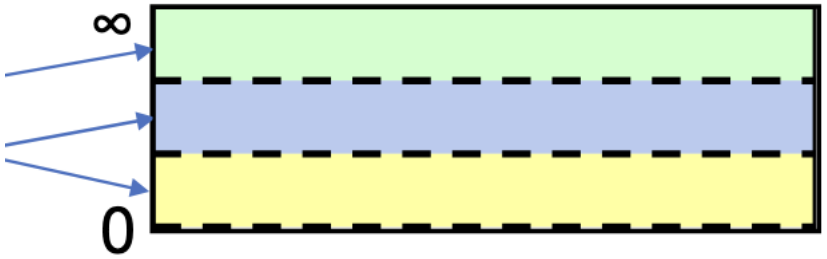
Keeping track of presence

- Our page map has a permissions bit and also a *present* bit.
- We must modify this every time that we move pages into our physical address space.

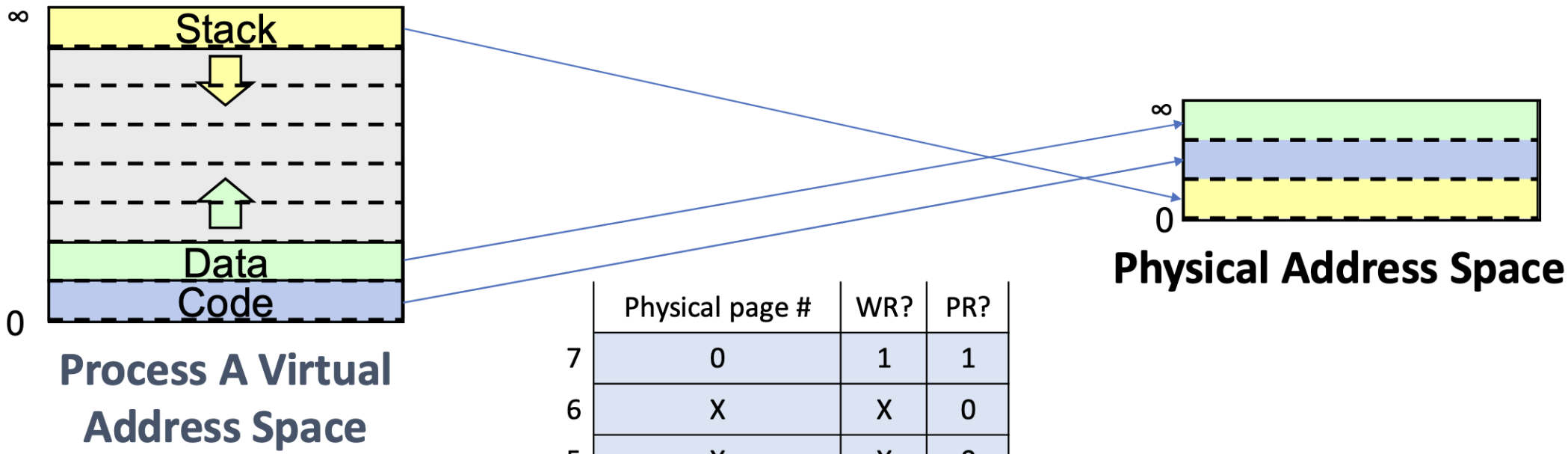
	Physical page #	WR?	PR?
7	0	1	1
6	X	X	0
5	X	X	0
4	X	X	0
3	X	X	0
2	X	X	0
1	2	0	1
0	1	0	1

In lecture

We only had three physical pages, so we had to use this technique track which pages were present in our page map

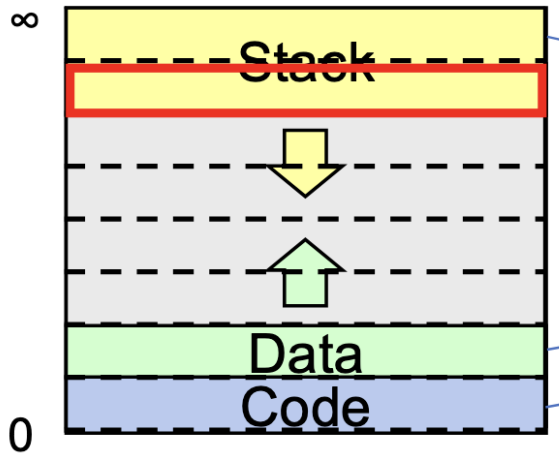


Let’s look at an example

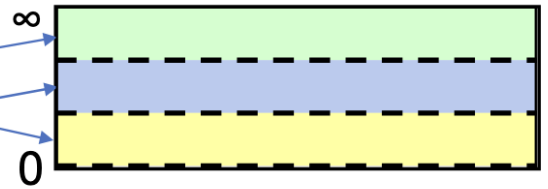


	Physical page #	WR?	PR?
7	0	1	1
6	X	X	0
5	X	X	0
4	X	X	0
3	X	X	0
2	X	X	0
1	2	0	1
0	1	0	1

From CS111 W26 lecture 23, slide 44



Process A Virtual Address Space

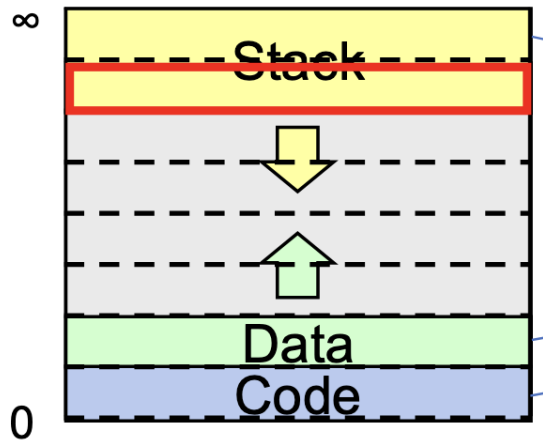


Physical Address Space

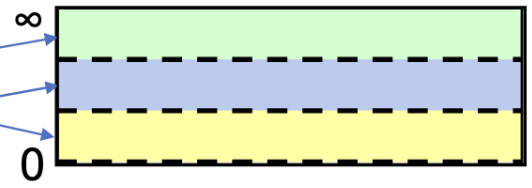
	Physical page #	WR?	PR?
7	0	1	1
6	X	X	0
5	X	X	0
4	X	X	0
3	X	X	0
2	X	X	0
1	2	0	1
0	1	0	1

1. Pick an existing physical page and swap it to disk.

From CS111 W26 lecture 23, slide 45



Process A Virtual Address Space

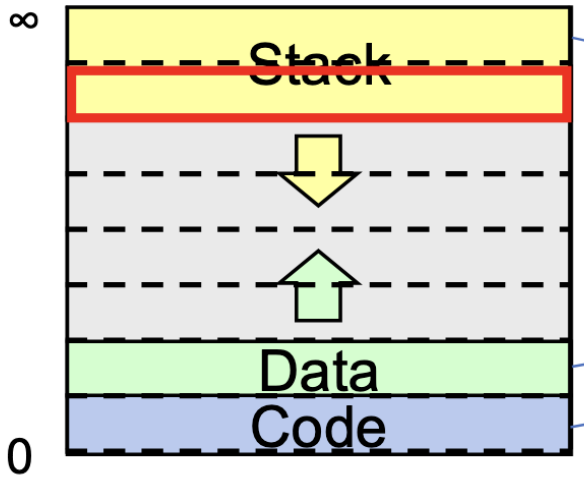


Physical Address Space

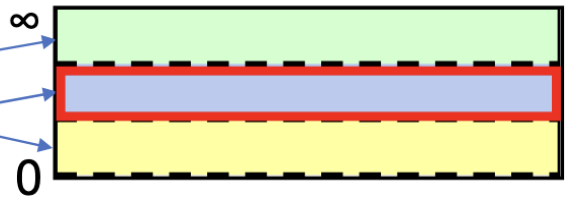
	Physical page #	WR?	PR?
7	0	1	1
6	X	X	0
5	X	X	0
4	X	X	0
3	X	X	0
2	X	X	0
1	2	0	1
0	1	0	1

1. Pick an existing physical page and swap it to disk.

From CS111 W26 lecture 23, slide 46

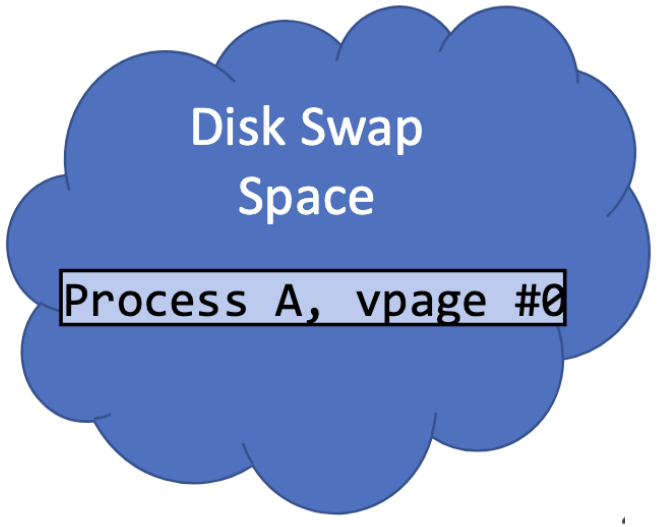


Process A Virtual Address Space



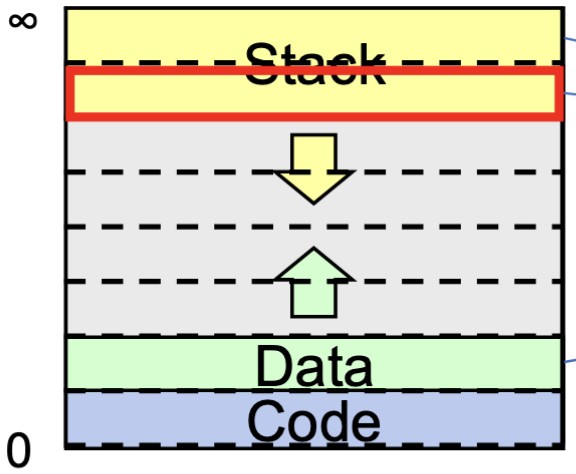
Physical Address Space

	Physical page #	WR?	PR?
7	0	1	1
6	X	X	0
5	X	X	0
4	X	X	0
3	X	X	0
2	X	X	0
1	2	0	1
0	1	0	0

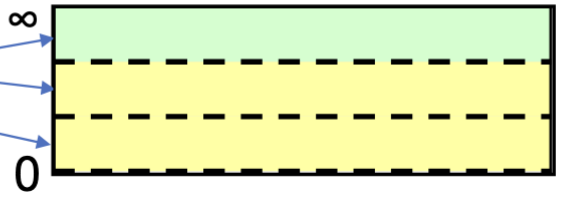


1. Pick an existing physical page and swap it to disk, mark not present.

From CS111 W26 lecture 23, slide 47

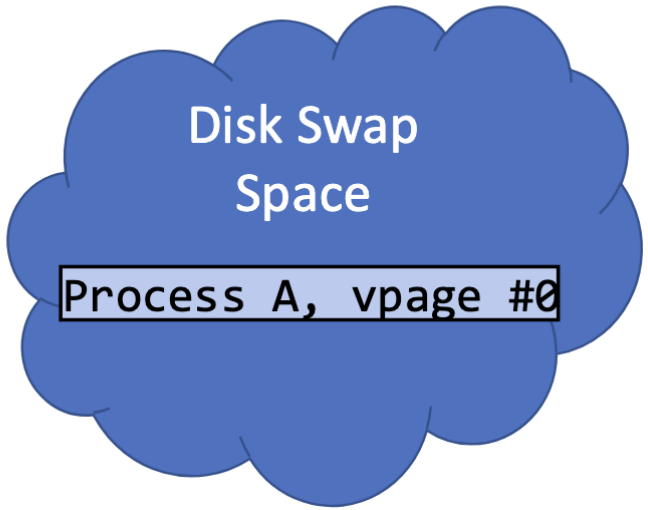


Process A Virtual Address Space



Physical Address Space

	Physical page #	WR?	PR?
7	0	1	1
6	1	1	1
5	X	X	0
4	X	X	0
3	X	X	0
2	X	X	0
1	2	0	1
0	1	0	0



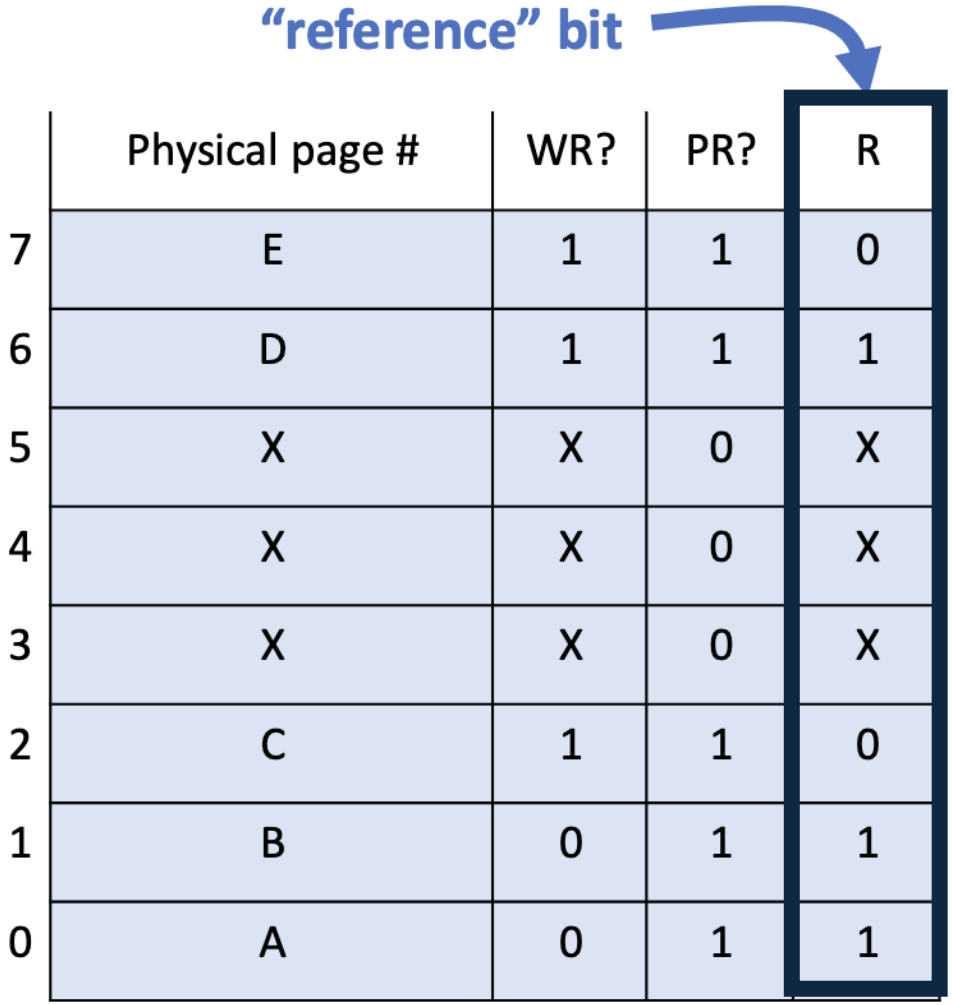
2. Map this physical page to the new virtual page.

From CS111 W26 lecture 23, slide 48

What questions can I answer?

How do you choose what page to swap?

The Clock Algorithm



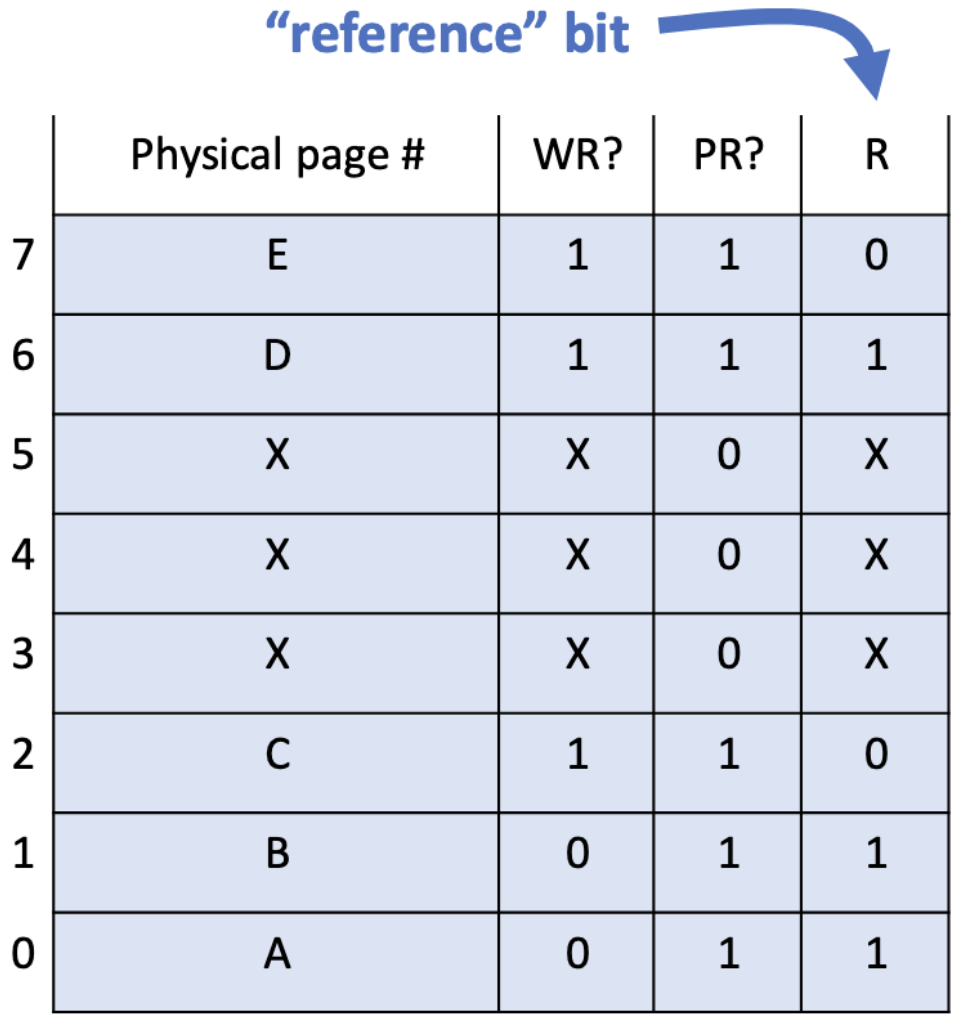
Set to 1 when read/referenced

If 1, when clock algorithm runs, set to 0

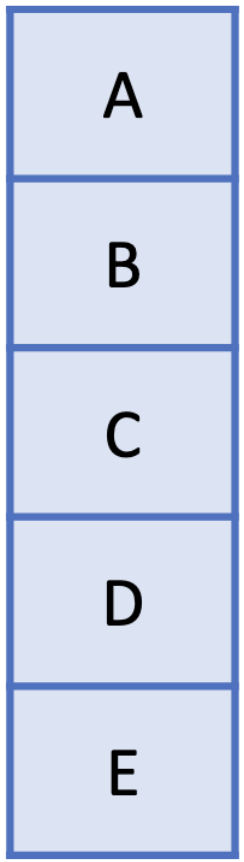
From CS111 W26 lecture 24, slide 19

The algorithm

- Scan through the pages
- If a page is 1 (it has been recently accessed/read), then mark as zero
- If a page is 0, then remove it



From CS111 W26 lecture 24, slide 19



Physical Pages

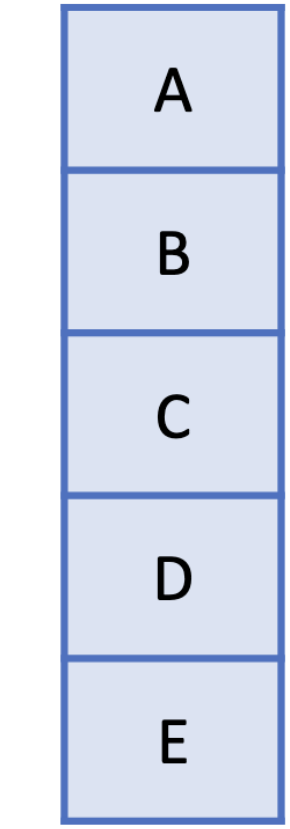
Let's say the system looks as follows, and a program requests mapping page 5, but we have no more physical pages. This triggers the clock algorithm.

"reference" bit

	Physical page #	WR?	PR?	R
7	E	1	1	0
6	D	1	1	1
5	X	X	0	X
4	X	X	0	X
3	X	X	0	X
2	C	1	1	0
1	B	0	1	1
0	A	0	1	1

Page Map

From CS111 W26 lecture 24, slide 20



Physical Pages

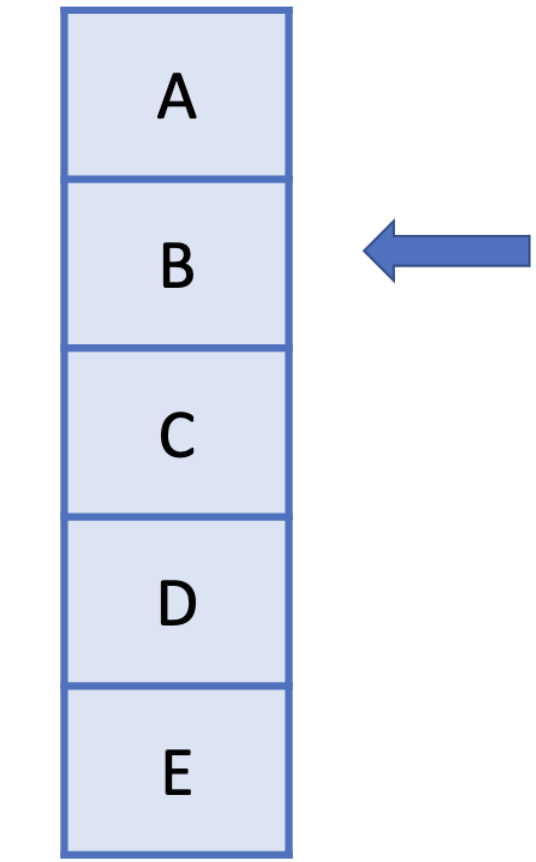
Was this page accessed recently (reference = 1)? If so, set reference = 0 and continue.

Have to set reference bit to 0!

"reference" bit

	Physical page #	WR?	PR?	R
7	E	1	1	0
6	D	1	1	1
5	X	X	0	X
4	X	X	0	X
3	X	X	0	X
2	C	1	1	0
1	B	0	1	1
0	A	0	1	0

Page Map



Physical Pages

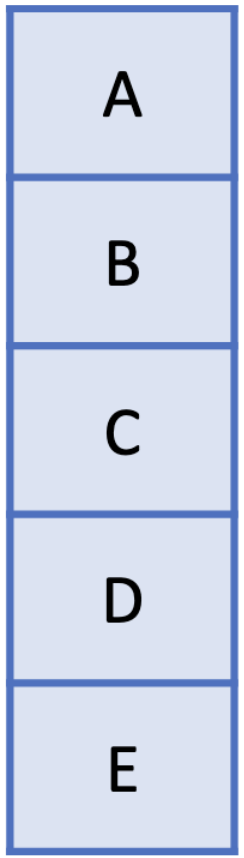
Was this page accessed recently (reference = 1)? If so, set reference = 0 and continue.

"reference" bit

	Physical page #	WR?	PR?	R
7	E	1	1	0
6	D	1	1	1
5	X	X	0	X
4	X	X	0	X
3	X	X	0	X
2	C	1	1	0
1	B	0	1	1
0	A	0	1	0

Page Map

From CS111 W26 lecture 24, slide 25



Physical Pages

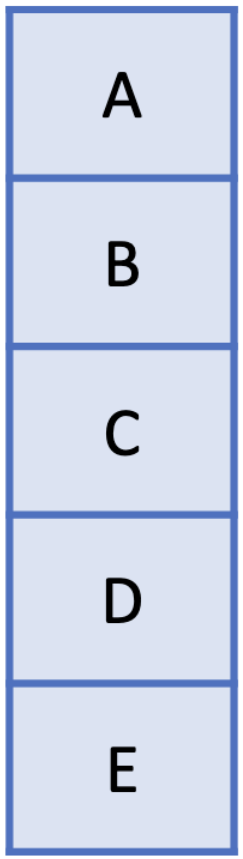
Was this page accessed recently (reference = 1)? If so, set reference = 0 and continue.

"reference" bit

	Physical page #	WR?	PR?	R
7	E	1	1	0
6	D	1	1	1
5	X	X	0	X
4	X	X	0	X
3	X	X	0	X
2	C	1	1	0
1	B	0	1	0
0	A	0	1	0

Page Map

From CS111 W26 lecture 24, slide 26



Physical Pages

Was this page accessed recently (reference = 1)? If not, this is the one we should remove.

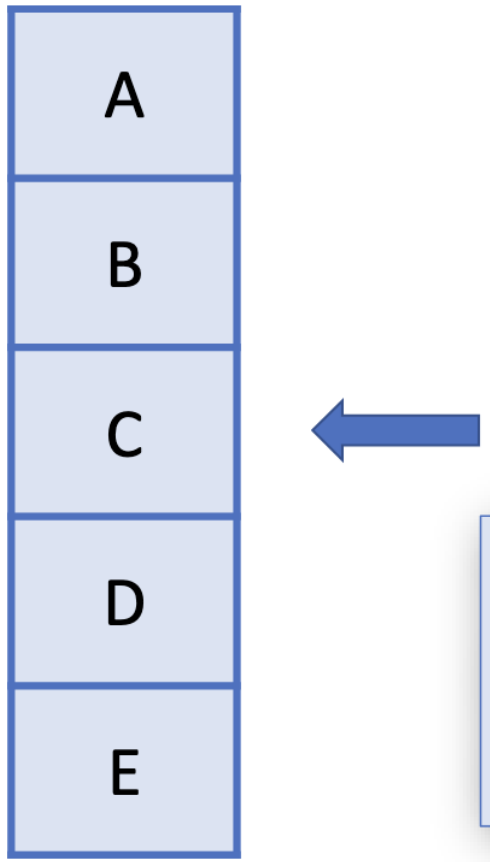
"This page hasn't been used 'recently' - let's remove it.

"reference" bit

	Physical page #	WR?	PR?	R
7	E	1	1	0
6	D	1	1	1
5	X	X	0	X
4	X	X	0	X
3	X	X	0	X
2	C	1	1	0
1	B	0	1	0
0	A	0	1	0

Page Map

From CS111 W26 lecture 24, slide 27



Physical Pages

Was this page accessed recently (reference = 1)? If not, this is the one we should remove.

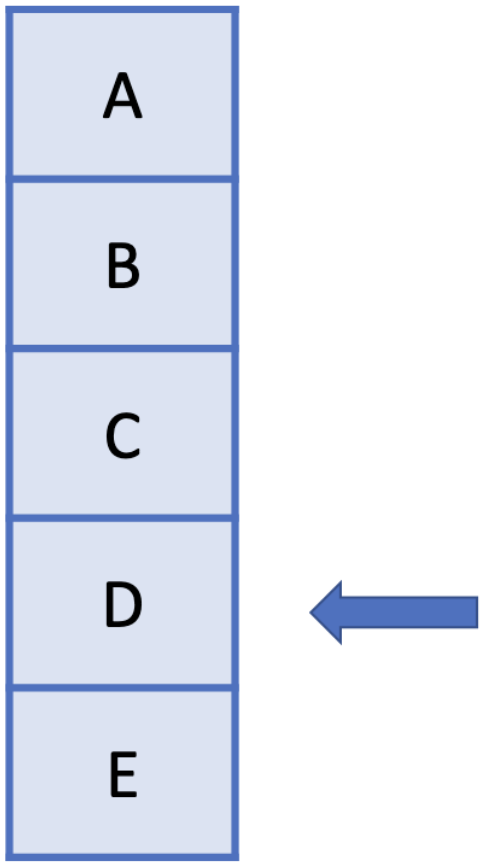
"This page hasn't been used 'recently' - let's remove it.

"reference" bit

	Physical page #	WR?	PR?	R
7	E	1	1	0
6	D	1	1	1
5	C	1	1	1
4	X	X	0	X
3	X	X	0	X
2	X	X	0	X
1	B	0	1	0
0	A	0	1	0

Page Map

From CS111 W26 lecture 24, slide 28



Physical Pages

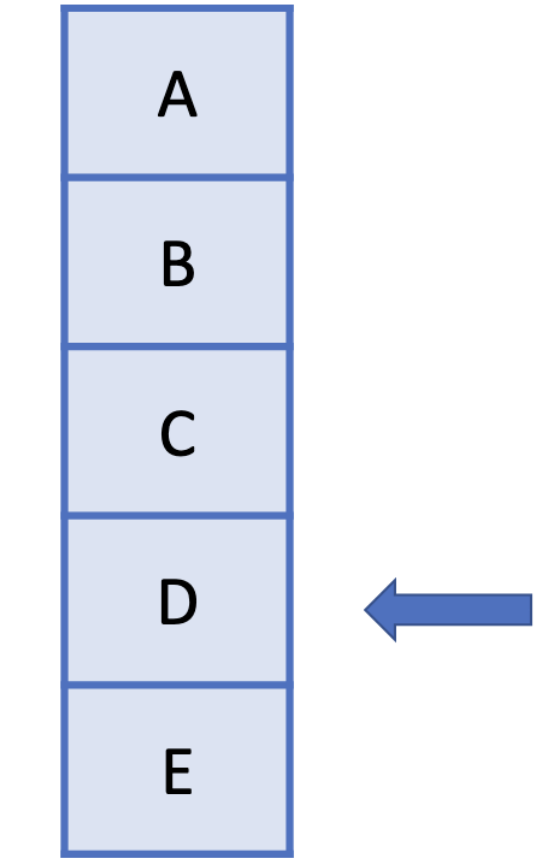
Now the clock algorithm stops, and **we remember where the hand left off for next time it runs.**

"reference" bit

	Physical page #	WR?	PR?	R
7	E	1	1	0
6	D	1	1	1
5	C	1	1	1
4	X	X	0	X
3	X	X	0	X
2	X	X	0	X
1	B	0	1	0
0	A	0	1	0

Page Map

From CS111 W26 lecture 24, slide 29



Physical Pages

In the meantime, the program resumes running, and a long time could pass between runs of the clock algorithm. During that time, pages could be accessed, meaning reference bits may change.

Reminder that the program can access/read these pages, which changes the reference bit value

"reference" bit

	Physical page #	WR?	PR?	R
7	E	1	1	1
6	D	1	1	1
5	C	1	1	1
4	X	X	0	X
3	X	X	0	X
2	X	X	0	X
1	B	0	1	1
0	A	0	1	0

Page Map

What questions can I answer?