1 Buffer Management

(a) Fill in the following tables for the given buffer replacement policies. You have 4 buffer pages, with the access pattern **A B C D A F A D G D G E D F**

Least Recently Used (LRU)

A														Hit Rate
	B													
		C												$\overline{14}$
			D											

Most Recently Used (MRU)

A														Hit Rate
	B													
		C												
			D											

CLOCK ("second chance LRU")

A														Hit Rate
	B													
		C												
			D											

(b) Fill in the following tables for the given buffer replacement policies. You have 4 buffer pages, with the access pattern **A, B, C, D, A, F (remains pinned), D, G, D, unpin F, G, E, D, F** Remember that unpinning does not contribute to the hit count!

Least Recently Used w/ Pinning

A															
	B														
		C													$\overline{13}$
			D												

Most Recently Used w/ Pinning

A														
	B													
		C												
			D											

(c) Is MRU ever better than LRU?

(d) Why would we use a clock replacement policy over LRU?

(e) Why would it be useful for a database management system to implement its own buffer replacement policy? Why shouldn't we just rely on the operating system?

2 Relational Algebra

Consider the schema:

```
Songs (song_id, song_name, album_id, weeks_in_top_40)
Artists(artist_id, artist_name, first_year_active)
Albums (album_id, album_name, artist_id,year_released,genre)
```

Write relational algebra expressions for the following queries:

(a) Find the name of the artists who have albums with a genre of either 'pop' or 'rock'.

(b) Find the name of the artists who have albums of genre 'pop' and 'rock'.

(c) Find the id of the artists who have albums of genre 'pop' or have spent over 10 weeks in the top 40.

(d) Find the names of all artists who do not have any albums.