SJSU CS 149 HW5 SPRING 2020

REMINDER: Each homework is individual. "Every single byte must come from you." Cut&paste from others is not allowed. Keep your answer and source code to yourself only - never post or share them to any site in any way.

[Type your answer. Hand-written answer is not acceptable.]

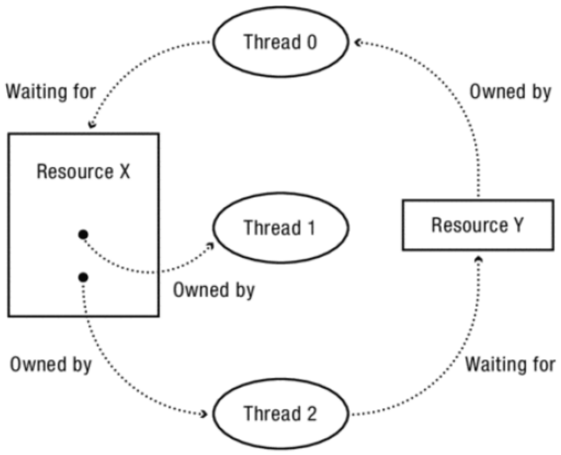
[Replace YourName and L3SID with your name and last three digit of your student ID, respectively.]

[This assignment does not have programming question, except the optional extra credit question.]

1. (15 pts) Given resource type X with two instances, resource type Y with a single instance, and three threads. As illustrated in the resource allocation graph,
   - Thread 0 owns resource type Y's instance and is waiting for resource type X.
   - Thread 1 owns one instance of resource type X.
   - Thread 2 owns the other instance of resource type X and is waiting for resource type Y.

a. (8 pts) Are Coffman conditions true in the current state represented by the resource allocation graph? must justify.

b. (7 pts) Is there any deadlock in the current state represented by the resource allocation graph? Why or why not?



2. (15 pts) Both segment table and page table are used to translate from logical address to physical address. But the structures of these tables are different; Each entry in a segment table is {limit, base} while each entry in a page table is {frame#}.

a. Why the differences?

b. Can we record base in a page table (and if so why we do not do it)?

c. Why there is no limit (or length) in a page table?

3. (30 pts) On a machine with 16 bytes page size, given the following page table for a process, and four of these 8 entries are mapped to page frames. Frame 0 starts at physical address 0. (All numbers given are in decimals.)

Page number	Frame number
0	2
1	0
2	Not in main memory
3	1
4	Not in main memory
5	Not in main memory
6	3
7	Not in main memory

a. (15 pts) Make a list of all logical address ranges (in decimals, byte-level) that would cause page faults.

b. (3 pts each) What are the corresponding physical addresses (in decimals, byte-level) of the following logical addresses (in decimals, byte-level)? If any address conversion is not possible, explain its reason.
   - 1
   - 50
   - 95
   - 96
   - 120

4. (40 pts) Consider the following page reference string: 7, 2, 3, 1, 2, 5, 3, 4, 6, 7, 7, 1, 0, 5
Assuming demand paging with 3 frames, fill in the table to indicate pages in the frames, page fault if any, and total number of page faults, for the following page replacement algorithms.

a. (20 pts) FIFO

Time	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Reference	7	2	3	1	2	5	3	4	6	7	7	1	0	5
Frame 0	7	7	7											
Frame 1	X	2	2											
Frame 2	X	X	3											
Page fault? (Y/N)														

Total page faults =_____________

b. (20 pts) LRU

Time	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Reference	7	2	3	1	2	5	3	4	6	7	7	1	0	5
Frame 0	7	7	7											
Frame 1	X	2	2											
Frame 2	X	X	3											
Page fault? (Y/N)														

Total page faults =_____________

Submit the following file:

- `CS149_HW5_YourName_L3SID` (.pdf, .doc, or .docx), which includes answers to all questions.

The ISA and/or instructor leave feedback to your homework as comments and/or annotated comment. To access annotated comment, click "view feedback" button. For details, see the following URL::

https://guides.instructure.com/m/4212/l/352349-how-do-i-view-annotation-feedback-comments-from-my-instructor-directly-in-my-assignment-submission

Optional - Extra credit (up to additional 15 points on top of 100 points)

5. Assume that a system has a 32-bit logical address with `N`-KB page size (where `N` >= 1, and 1KB = 1024 bytes). Write a C program that accepts two command line parameters, the first one being a logical address in decimal notation and the second one being the value of `N` (in decimal notation). The program outputs the logical address, the page size, the page number and offset for the given logical address. For example,

```
./logicaladdr 19985 4
```

The program should output

```
Logical address translation by <YourName> <L3SID>
logical address 19985, page size = 4096 => page number = 4, offset = 3601
```

Replace YourName and L3SID with your own name, and last 3 digits of your SID. Test your program with the following **four** runs

```
./logicaladdr 15991 1
./logicaladdr 15992 2
./logicaladdr 15994 4
./logicaladdr 15998 8
```

and capture screenshots of your program execution.

Submission:

- At the end of the regular report, include screenshots of those four runs. Note each screenshot must include "Logical address translation by …".
- Your source code, named as `logicaladdr_<YourName>_<L3SID>.c`