89/100

COSC 450 Operating System Midterm #1-Again

(77+89)/2 = 83

10/26/2022

Name: Jung An

1. Short answer questions
   a) (1 pt.) Modern operating system has three types of schedulers: long term, short term and memory scheduler. Why the second generation operating system does not have short term scheduler? The second generation OS does not support multiprograming so it does not have short term scheduler.
   b) (1 pt.) What are two main functions of OS for supporting multiprogramming?
      a. protection of job
      b. schedule jobs
   c) (1 pt.) What is race condition between processes?
      race condition is when two or more processes try to access same memory at the same time. and ?
   d) (1 pt.) Why effective short term scheduling algorithm is essential?
      It is essential so that the processes do not have deadlock / Mutex violations
   e) (1 pt.) A major problem with priority scheduling algorithm used in a scheduler is starvation. What is the solution of starvation?
      The solution of starvation is aging which increases priority depending on time in waiting.
   f) (1 pt.) What is the motivation of virtual memory?
      The virtual memory is used to reduce amount of data being stored in physical memory.
   g) (1 pt.) One of design issue of paging system is page size. Discuss disadvantages for smaller page size and bigger page size.
      big page - expansive, longer search time
      Smaller page - large bitmap
   h) (1 pt.) processes can use message queue or shared memory for inter-process communication. What is the main role of operating system for message queue and shared memory?
      OS syncs the messages for the message queue while shared memory has to do it on it's own.
   i) (1 pt.) When a system uses virtual memory, virtual address do not directly onto the memory bus. Instead, it goes to an (MMU) that maps onto the physical memory address.
   j) (1 pt.) Main goal of batch system for selecting proper scheduling algorithm are maximize throughput, minimize the turnaround time and CPU utilization. what are throughput and turnaround time?
      a. Throughput - Amount of process per unit time
      b. Turnaround time - Total Amount the process takes to complete.

2. (10 pt.) A computer system generates a 32-bit virtual address for a process. This system has 8 GB RAM and page size is 8KB.
    - If each entry of the page table needs 64 bits per entry, calculate the possible size of the page table by bytes.

Page size: $2^{32} / 2^{13} = 2^{19}$

$2^{19} \cdot 2^{6} = 2^{25}$ bits $\quad 2^{25}/2^{3} = 2^{22}$ bytes

    - Page frame number information for each page must be saved in the page table. **How many bits** does it need to save page frame number information?

# of frames $= 2^{3} \cdot 2^{30} / 2^{3} \cdot 2^{10} = 2^{20}$

it requre 20 bits

3. (10 pt.) Mr. Computer tries to solve the race condition problem in the producer-consumer problem. He comes up with following solution. Are there any problems with his solution? If his method solves the race condition, show it. If his method could not solve the race condition, write a scenario lead to the race condition situation.

```
#define N 100
int count = 0;
void producer()
{
        int item
        while (ture)
        {
                item = produce_item();
                if (count == N)
                   sleep();
                insert_item(item)
                count = count + 1;
                if (count ==1)
                   wakeup(consumer);
        }
}
```

```
void consumer()
{
        int item;
        while(true)
        {
                if (count == 0)
                   sleep();
                item = remove_item();
                count = count – 1;
                if (count == N – 1)
                  wakeup(producer);
                consume_item(item);
        }
}
```

Producer is called when count == N and it reads N and times out

Then consumer is ran multiple times until count == 0 but does not wake producer since it is not sleep

Then producer continues and since it read N before it goes to sleep

Since count == 0 at this point consumer also go to sleep

Both producer and consumer sleeps forever

4. (10 pt.) A computer can generate 32 bit virtual address for a process. This system has 4GB RAM and page size is 4KB.
    - Let's assume page #0 is map to page frame #23
    - Let's assume page #1 is map to page from #5
    - Let's assume page #2 is map to page frame #1100
    - Let's assume page #3 is map to page frame #1230
    - Let's assume page #4 is map to page frame #135
    - Let's assume page #5 is map to page frame #95
    - Let's assume page #11 is map to page frame #7.

Calculate physical address for each of following virtual addresses (need show the sequence of caluation)

a) 19845

page # = $19845 / 2^{12} = 4$ page #4 map to frame 135

offset = $19845 - (4 \cdot 2^{12}) = 3461$

Physical address = $(135 \cdot 4 \cdot 2^{10}) + 3461 = 556421$

b) 21581

Page # = $21581 / 2^{12} = 5$

offset = $21581 - (5 \cdot 2^{12}) = 1101$ page # 5 map to frame 95

physical address = $(95 \cdot 4 \cdot 2^{10}) + 1101 = 390221$

5. (7 pt.)Probabilistic model for multiprogramming.

a) Lets *p* is an average fraction of time a process is waiting for I/O. If there are n processes in the memory at once, what will be the CPU utilization?

$1 - p^n$

b) A computer has 8 GB of memory, with operating system taking up 1GB and each program taking up 1 GB. With an 90 % average I/O wait, calculate CPU utilization

$n = (8GB - 1GB)/1GB = 7$

Cpu util = $1 - .9^7 = .5217 = 52.17\%$

c) By doubling the size of memory, what will be the CPU utilization?

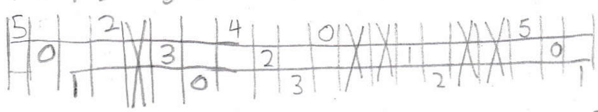
$n = (16GB - 1GB)/1GB = 15$

CPU util = $1 - .9^{15} = .7941 = 79.41\%$

6. (10 pt.)How many page faults occur for following reference string with three page frame for each of following page replacement algorithm? Let's assume page frames are initially empty.
Reference string: 5, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 5, 0, 1

a) FIFO :

5 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 5 0 1

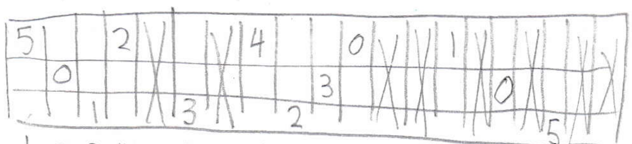


1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

15 page fault

b) LRU:

5 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 5 0 1



1 2 3 4 5 6 7 8 9 10 11 12

12 page fault

c) Optimal:

5 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 5 0 1

1 2 3 4 5 6 7 8 9

9 page fault

-5

7. (10 pt.) A Computer with a 64 bit virtual address use two-level page table. Virtual addresses are split into a 25 bit top-level page table field, a 25 bit second-level page table field and an offset. (need check) 64 bit (25 25 14)

- What is the size of each page ?
  2^{14} = 16KB Since offset itself is a page, we can use that as page size
- How many possible page tables are there?
  $1 + 2^{50}$ the offset has a table + 25 top level + 25 second level
- How many possible pages are there?
  2^{50} page $\cdot$ 2^{50} entry $= 2^{100}$ pages
- If the system supports 8GB RAM, how many bits need to be reserved for saving page frame number in the each of page table entry?
  $2^3 \cdot 2^{30} / 2^4 \cdot 2^{10} = 2^{19}$

  19 bits

8. (8 pt.)Mr. Computer tries to solve the race condition problem in the producer-consumer problem by using semaphores. He comes up with following solutions. His solution leads to a deadlock situation (both producer consumer go to sleep forever). Write a scenario which lead to a deadlock for his solution.

```
#define N 100
typedef int semaphore;
semaphore mutex = 1;
semaphore empty = N;
semaphore full = 0;

void producer ()
{
        int item;
        while (ture)
        {
                item = produce_item();
                down (&mutex);
                down (&empty);
                insert_item(item);
                up(&mutex);
                up(&full);
        }
}
```

```
void consumer()
{
        int item;

        while (true)
        {
                down(&full);
                down(&mutex);
                item = remove_item();
                up(&mutex);
                up(&empty);
                consume_item(item);
        }
}
```

Assume that consumer was scheduled consecutively and empty = 0
Then producer is called and locks the mutex. It then tries to down empty ~~but~~ Since it is 0, it waits
Then consumer is called and downs full Then tries to go to mutex but its locked so it waits.
At this point both producer and consumer sleeps forever.

9. (5 pt.) For a decimal virtual address 45250, compute the virtual page number and offset for a 2 KB and for a 4 KB page size in paging system.

2kB = page# $= 45250/2^{11} = 22$ offset $= 45250-(22 \cdot 2^{11}) = 194$

4kB = page# $= 45250/2^{12} = 11$ offset $= 45250-(11 \cdot 2^{12}) = 194$

10. (10 pt.) You are going to compare the storage space needed to keep track of free memory using a bitmap versus using a linked list. The 2 GB memory is allocated in units of 4KB. For the linked list, let's assume that memory is currently consists of an alternating sequence of segment and holes, each 128 KB. Also assume that each node in the linked list needs a 32-bit memory address, a 16-bit length, and a 16-bit next node field. How many bytes of storage are required for each method?

Sol)

- Bitmap:

in bit: $2^{31}/2^{12} = 2^{19}$

Convert to Bytes: $2^{19}/2^{3} = 2^{16}$ Bytes

- The linked list:

nodes: $2^{31}/2^{17} = 2^{14}$ nodes

#bits per node = $32+16+16 = 64$ bits

linked list bit $= 2^{14} \cdot 2^{6} = 2^{20}$

convert to bytes: $2^{20}/2^{3} = 2^{17}$ Bytes

11. (10 pt.) Multiple jobs can run in parallel and finish faster than if they had run sequentially. Suppose that two jobs, each needing 20 minutes of CPU time, start simultaneously. How long will the last one take to complete if they run sequentially? How long if they run in parallel? Assume 50% I/O wait.

a. Sequential: since it takes 20 minute for each,

20 min + 20 min = 40 min. X 40 + 40 = 80

b. parallel: n: 2 process P = 50% CPU util = $1 - .5^2 = .75$ for both

$\frac{20\text{ min}}{.75} + \frac{20\text{ min}}{.75} = 53.33$ min