Daniel Frey
CS 4500-001
Homework 2
5/5/19

I have neither given nor received unauthorized assistance on this work
Name: Daniel Frey *Date:* 5/5/19

Chapter 3:

1. Segmentation is for breaking up data and programs into logically independent address spaces. This helps to aid sharing and protection. A segment table also takes up less space than a paging table.

2. Compute virtual page number and offset, 2KB page and 4KB page; 16000, 8192, 30000
   Calculations applied: Page # = (Addr/Size(B)), Offset = Addr - Page# × Size(B)
   2KB:
   16000: Page # = 7, Offset = 1664
   8192: Page # = 4, Offset = 0
   30000: Page # = 14, Offset = 1328
   8KB:
   16000: Page # = 3, Offset = 3712
   8192: Page # = 2, Offset = 0
   30000: Page # = 7, Offset = 1328

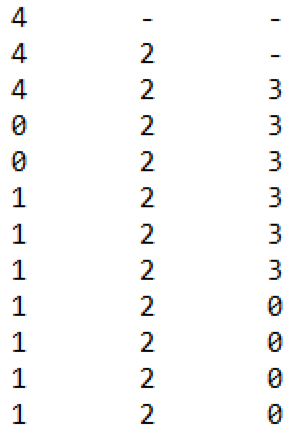
3. Two-level page table, 4KB pages, each level 10-bits, decimal VA if PT=5, PT=2, offset=3
   ~~$(2^5+2^2+2^3)\times4$ = 176B page table size~~
   $(5\times2^{22})+(2\times2^{12})+3=20979715$

4. The clock page replacement algorithm works by having pages ordered in a circular list where the hand points to the oldest page first. Each page uses a reference, R, bit to determine if the page had recently been referenced. When a page fault occurs, the page that the hand points to is inspected. The action taken is determined from the R bit; if the R bit is 0, then the page is evicted; if the R bit is 1, then R is cleared, and the hand is advanced to the next page.

5. External fragmentation are the holes of free space between segments. External fragmented blocks can be allocated, but the sizes of the blocks may be too small to be efficiently used. Internal fragmentation is the wasted space in allocated blocks, which is more difficult to reclaim. This results from inefficient memory usage.

6. 3 physical frames of memory, sequence: 4 2 3 0 2 1 2 3 0 1 2 1, # page faults

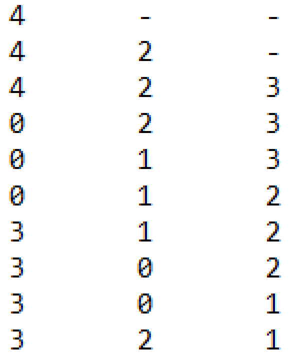
a. Optimal Page Replacement Algorithm



```
4       -       -
4       2       -
4       2       3
0       2       3
0       2       3
1       2       3
1       2       3
1       2       3
1       2       0
1       2       0
1       2       0
1       2       0
```

# page faults = 6

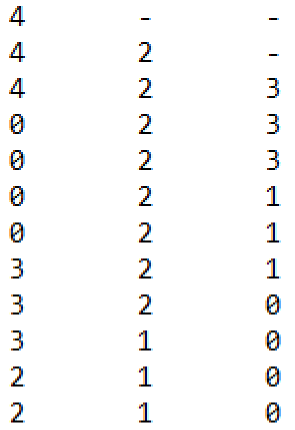
b. FIFO



```
4       -       -
4       2       -
4       2       3
0       2       3
0       1       3
0       1       2
3       1       2
3       0       2
3       0       1
3       2       1
```

# page faults = 10

c. LRU



```
4       -       -
4       2       -
4       2       3
0       2       3
0       2       3
0       2       1
0       2       1
3       2       1
3       2       0
3       1       0
2       1       0
2       1       0
```

# page faults = 9

7. VM system, 16-bit virtual address, page size 256 bytes, single level page table, # bits used to represent page offset and virtual page number. Assuming 4 bytes per page table entry--size of page table.
   $256 = 2^8$ → 8 bits in offset
   16bits – 8 bits = 8 bits in virtual page number
   $2^8 \times 4 = 2^{10}$ size of page table

Chapter 4:

1. An advantage of byte sequence file structure is that the operating system does not need to know what is in the file, it simplifies file management. A disadvantage is that any meaning must be imparted by the user programs.
   An advantage of record sequence is that reads return one whole record of data so there are less reads (same for writes). The operating system can optimize record operations. A disadvantage is that you may be reading more information than necessary, therefore there remains unused portions.
   An advantage of tree is that it allows for fast searching for a particular key. The disadvantage is that you must have the whole tree in order to find something, so it will take up more space.

2. A memory-mapped file is in virtual memory with the contents of the file, it is a direct byte-for-byte correlation. The advantages of memory-mapped files are that it increases I/O performance since accessing local virtual memory is faster, and processes mapped onto the same file used shared memory space to communicate.

3. Advantages of contiguous allocation are that it is simple addressing and has one seek only reading. A disadvantage is that the space becomes fragmented, like Swapping.
   An advantage of linked list allocation is that no space is lost due to fragmentation, aside from the last block. A disadvantage is the slow random access because of the linked list structure.
   Advantages of FAT are that there is high utilization and random access is easy. The main disadvantage is that it requires a decent amount of disk space.

4. I-node requires much less space than FAT because I-nodes are only a certain number of *n* bytes for *k* files, thus requiring *kn* bytes. FAT, however, is proportional to the disk, so a disk with *m* blocks requires a table of *m* entries.

5. A symbolic link is a reference to a file or directory. It contains the path to the linked file. Removing a symbolic link does not affect the actual file.

6. Tradeoffs between a large disk block size and a small disk block size occur in space utilization and data rate. A large disk block size will waste space, but a small disk block size will waste time.

7. Free list-based contains a list of the free blocks in a linked list, where a free block contains a pointer to the next free block. One drawback to this is the time it takes to traverse the list. Bitmap-based is a collection of bits that correspond to a disk block. Advantages to this are it's simple to understand, and it is faster to find free blocks.

Chapter 5:

1. Two examples of block devices are hard disks and optical drives (like Blu-ray). Two examples of character devices are printers and network interfaces.

2. 300 cyls – 0-299, request at 143, previous was 15, pending requests FIFO order: 86, 147, 291, 18, 95, 151, 175, 30; starting from current, total distance traveled
   a. FCFS
      143→86→147→291→18→95→151→175→30
      57+61+144+273+77+56+24+145 = 837

b. SSF
143→147→151→175→95→86→30→18→291
4+4+24+80+9+56+12+273 = 462

c. Elevator Algorithm
143→147→151→175→291→95→86→30→18
4+4+24+116+196+9+56+12 = 421

Additional Problems:

1. Processes A-E, estimated running times: 10, 6, 2, 7, 8 ms; arrival order A to E: 0, 1, 2, 3, 4 ms; determine avg turnaround and response time
    a. Round Robin, quantum = 4ms
    Turnaround time: ((33-0)+(24-1)+(10-2)+(27-3)+(31-4))/5 = 23
    Response time: ((0-0)+(4-1)+(8-2)+(10-3)+(14-4))/5 = 5.2
    b. FCFS
    Turnaround time: ((10-0)+(16-1)+(18-2)+(25-3)+(33-4))/5 = 18.4
    Response time: ((0-0)+(10-1)+(16-2)+(18-3)+(25-4))/5 = 11.8
    c. SJF
        i. Preemptive
        Turnaround time: ((33-0)+(9-1)+(4-2)+(16-3)+(24-4))/5 = 15.2
        Response time: ((0-0)+(4-1)+(2-2)+(9-3)+(16-4))/5 = 4.2
        ii. Non-preemptive
        Turnaround time: ((10-0)+(18-1)+(12-2)+(25-3)+(33-4))/5 = 17.6
        Response time: ((0-0)+(12-1)+(10-2)+(18-3)+(25-4))/5 = 11