Matt Buland
CSCI 442 Homework 5

7.2

The capability to relocate a process can be helpful for CPU scheduling, and also for security. Relocating a process can make it harder to hack in to.

7.3

It's not possible to enforce memory protection at compile time because the compiler is never sure of where the process will be stored when it's ran. Can't protect memory if you don't know where the memory is.

7.4

Two or more processes may have access to a particular region of memory if it is being offered as a public library, such as a shared object file, which once loaded into memory, may be used by many programs. Sharing this library amongs many processes will save memory.

7.5

The advantages of unequal partitions is greater variety. Since there are more choices in location size, processes can be better positioned, reducing external fragmentation.

7.6

Internal fragmentation is a split of process memory within a partition, external is a split of memory segment across many partitions

7.8

A page is a small segement of memory that exists within a frame. Frames are virtual memory segments of uniform size. The base address of a frame can be found using a look-up table.

7.11

Maintaining a relative address in the instruction register is preferrable. Since it's common to be able to access and change the program counter (jumps), it needs to be readily accessible. Holding an absolute address in there is a security risk, as it unviels the current position in memory that the program is stored. Since going through an adder is required to perform the jump, another circuit can bounds-control the jump, denying access to unauthorized programs. This would not work so well with an absolute program counter.

7.12

2^32 physical memory, page = 2^10, 2^16 pages of logical address space

a. 2^10 * 2 ^ 16 = 2*(26+1) = 134217728 bytes

b. 2^32/2^26 = 2^8 frames...?

Q.7.2

Critique of Fixed Partitioning

con: Large jobs may become very fragmented

con: Partition sizes are preset, and therefore may not be well suited for the tasks at hand

Assuming that best available == first fit...

Best fit will be better with fragmentation, since it maximizes the amount of available memory for other processes, allowing for more, or larger jobs. It tends to be much slower, but best fit will provide less fragmentation.