**Tran Nguyen**

**CS 3343 Operating Systems**

**Assignment 8 10 points**

**Chapters 8 and 9**

**Due 4/27**

**Email your homework to me at** [**harringp@nsuok.edu**](mailto:harringp@nsuok.edu)

**Use the slides or textbook definitions to answer the following (12 points)**

1. What do the base and limit registers on a cpu define?

Base register contains value of smallest physical address. Limit register specifies the size of the range. A pair of base and limit registers specifies the logical address space and can be loaded in operating system.

1. What three possible times of software creation can memory have instructions and data bound to it?

Compile time: If memory location known a priori, absolute code can be generated; must recompile code if starting location changes.

Load time: Must generate relocatable code if memory location is not known at compile time.

Execution time: Binding delayed until run time if the process can be moved during its execution from one memory segment to another. Need hardware support for address maps.

1. What is the difference between logical and physical addresses?

The basic difference between a logical address and a physical address is that a logical address is generated by the CPU from the point of view of a program. On the other hand, a physical address is a location that exists in a memory unit.

Logical addresses are also called virtual addresses because logical addresses do not physically exist in a memory unit. The physical address is a location in a unit of memory that can be physically accessed.

1. What is the hardware that maps virtual to physical addresses?

The memory management unit of the CPU contains a relocation register. Whenever a thread tries to access a memory location, the value of the relocation register is added to the virtual memory address. The kernel maintains a separate relocation value for each process, changes the relocation register at every context switch.

1. What is the difference between dynamic loading and linking?

dynamic loading:

Routine is not loaded until it is called

Better memory-space utilization; unused routine is never loaded

Useful when large amounts of code are needed to handle infrequently occurring cases

No special support from the operating system is required implemented through program design

dynamic linking:

Linking postponed until execution time

Small piece of code, stub, used to locate the appropriate memory-resident library routine

Stub replaces itself with the address of the routine, and executes the routine

Operating system needed to check if routine is in processes’ memory address

Dynamic linking is particularly useful for libraries, system also known as shared libraries

1. What are the two hardware components needed for contiguous memory allocation?

Main memory and relocation register

1. Describe the two types of fragmentation.

Both internal and external fragmentation are natural related to unused or wasted memory space. Fragmentation inside inefficient memory allocation occurs when memory is allocated to a process more than what is required, leaving unused space in the memory block eventually causing the allocation. piece inside. After the process is removed from physical memory, the free space will be distributed here and there and no adjacent block of memory can be found, causing external fragmentation. Internal fragmentation can be reduced by allocating memory to processes flexibly, while external fragmentation can best be avoided by compression, paging, and segmentation.

1. What are the two parts of an address generated by a cpu to handle paging?

A page number  
An offset

1. When does a page fault happen?

Page error occurs when a user visits an invalid page, meaning that the page is not in the process address space is not in main memory, or we are accessing a valid page but is in second memory.

1. What is thrashing?

Thrashing in computing is an issue caused when virtual memory is in use. It occurs when the virtual memory of a computer is rapidly exchanging data for data on hard disk, to the exclusion of most application-level processing. As the main memory gets filled, additional pages need to be swapped in and out of virtual memory. The swapping causes a very high rate of hard disk access.