**Assignment 3: Memory Management**

In this coding assignment, we explore the methods of simulation of memory management policies in a limited resource and size. Our objective is that we read some input files that contains some workload data and then simulate the execution of the processes by the memory manager. The program will generate an output file with memory map and input queue status with a turnaround time data output.

Pseudo Code:

**Block Check:**

Int available(vector MM, int size)

{

For (int i = 0; i < MM.size(); ++i){

//check if free

//If free, temp ++

//If not, return 0

//if Temp = size, val = (i+1-size)

Return val;

**Importing input:**

For (int i = 0; i < input.block; ++i)

{

//Loop Size[i]

//Loop size[j]

//Into available MM

For (int i = 0; i < MM.size(); ++i)

{

//If MM is free, increment num

//If num = Size, space = (i +1 - Size)

//If MMstart > -1, leave it alone

}

}

**Output**

//Loop and display the output

For (int i = 0; i < (memSize / pageSize); ++i){

//Initialize MM

}

For (int i = 0; i < order.size(); ++i){

//output t

For (int j=0; j < list.size(); ++j){

//output queue and time

}

}

//Display memory map

//Add and dequeue process to memory map

**Main Function:**

Int main(){

//error check command line arguement

//Read from file code

//Store values for read and vector into size class object

//Calculate the order

}