CSCD 340
Assignment 4

PROGRAM SPECIFICATIONS

We model the memory managed in a operating system as a list of memory blocks. Each block of memory is modeled by an object of MemoryBlock structure defined below:

```
struct MemoryBlock
{
	int startAddress;
	int endAddress;
	int segmentSize;
	int processId; //0 indicates a free block
};
```

Thus, the operating system code can set up memory blocks for the available memory and keep track of which blocks are allocated to which processes. In particular, note that for each memory block, the system tracks the starting and ending addresses, along with the size of the block and the process to which the block is currently allocated. For free blocks, the process is indicated as a zero (0).

The system manages the memory as a list of memory blocks. As memory is allocated and released, the list of blocks in the memory map changes.

In lab 9 we hand traced the code to understand the concepts. Then we wrote a short compare and contrast based on what we believe will happen. In this homework we will prove we were correct/incorrect. The difference is we will also add placement timings to further compare the algorithms based on the placement time.

Note that if there is no free block of memory (in the memory map) that is at least as large as the requested size, you must gracefully handle this situation and report it to the user. If the free block found is larger than the requested size, the block is split into two pieces - the first piece allocated and the second piece becoming a free block in the memory map.

OTHER SPECIFICATIONS

Your task is to write a C program that codes the 4 algorithms (Best Fit, First Fit, Worst Fit, and Next Fit) from lab 9 with the following specifications. You must implement First-Fit and 2 other algorithms. (You don't need to implement all 4 and no you won't get extra credit if you do)

- There an initial text file named initialMemory.txt that contains the starting memory map. Each item will be on a separate line. You will need to read the file and build the initial memory map. The very first line in the file will be the total size of memory and the starting address.

  Example (Type, ID, Size, Start)
  400 112
  P 1 40 172
  P 2 12 224
  P 3 32 236

- You will have some kind of user interaction that allows for the transactions, similar to what we saw in lab 9.

- Create a simulation that allows for the following:
  a. Initial Memory Map Set Up
  b. Printing of the memory map at any time. Your printout will indicate the process id, the size and the start and ending locations. Holes should be clearly marked.
  c. You must use some kind of times and keep track of the placement time. Example user enters new process P4 40 278 – you will keep track based on each algorithm of how long it took to place that new process.

- Name your file that contains main cscd340hw4.c.

- You must provide at least a single page write up, saved as a PDF that does a thoughtful analysis of your simulation with a comparison of processes and fragmentation. Meaning for the algorithms you will indicate which was better placing the processes; however by placing the processes according to the algorithm then how did that effect fragmentation. Your write-up will include a graph of the information of:

  - The number of fragments per algorithm
  - The average size of the fragments per algorithm
  - The average placement time per algorithm

- This graph will be on a separate page and does not count as the page write-up.
- Please don't "half-ass" this write-up. It is expected that you will create a thoughtful analysis. This analysis will account for a large portion of your grade. It will be single spaced and no larger than 12 point font with 1 inch left and right margins and 1 inch top and bottom margins.

TO TURN IN

A zip file that contains:

- All C files to ensure your code properly compiles and runs
- A simple Makefile, with the target hw4
- Your input files
- A PDF of your analysis, including the graphs

Name your zip file your last name first letter of your first name hw4 (Example steinershw4.zip)

DON’T WAIT TO GET STARTED