Read me

Yusdel Lima Lorenzo

2/18/2018

The malloc function I created acts as a first fit function and my implementation of free merges freed space to optimize space. With the use of a struct Node I was able to implement a linked list of nodes which comprise of the total memory. Within the struct itself is two pointers next and previous which are used to traverse the memory block. Also, in the struct is the variable isFree which is an indicator if the malloced block is free or not. Lastly, is the size variable which holds the size requested by the user. I decided to assign this variable to a unsigned short because the maximum value allowed was 5000. When it comes to errors I used the variables file and line to prompt the user with specific locations to the errors. One interesting thing that I ran into was on workload B the loop was never reaching the second workload therefore I used the linux command >> to output the result of the terminal to a file named “testB.txt”. I outputted the result of the function 100 times and then averaged the numbers. Lastly, the makefile makes sure the project is properly linked together. For workLoad A, E, and F the average time it took was around 0.004 milliseconds and when used in an array such as in workload B it took 0.08 milliseconds. This is because the malloc and free function has to go through the array one by one.