Tomer Levy and Neil Malu (tl455 & nmm182) 157003722 & 167009898

Asst1: Memory Allocation ++

For our mymalloc we use an array that uses 2 bytes of meta-data, with a simple design, if the meta data is a positive number that means that the byte specified in the meta data is the size of free memory pointed to by the adjacent element in the array. If the meta data is negative it symbolizes that (-1) times the meta data is the size of the memory pointed to by the adjacent element in the array. It handles exceptional cases, anything less than 0 will give an input error anything greater than 5000 will print out an error saying there is not enough space with file name and line number of the function call. All errors include this feature. We have a cleaning up method in which it takes contiguous blocks of free data and converts them into one block to reduce the amount of meta-data present. There were many issues while writing this code, most of which had to do with pointer arithmetic and type casting. For example: typecasting from char* to short. Most of the time we spent on this assignment was researching how pointers actually work and why we couldn't use simple typecasting without getting tons of errors. Our malloc efficiently runs the workloads given to us, and the ones we created. These operations are all being done in microseconds.