Arifur Rahman
CS 214- Section 5

This read me explains my implementation of the malloc and free functions. They use an already allotted amount of memory, which is 5000 bytes of memory in the code. This can be changed by setting the MAXSIZE macro in the othermalloc.h header file. Everything will adjust so it works properly.

My implementation handles fragmentation by splitting into two chunks of memory: a chunk for larger requests, and a chunk for smaller requests. This is set in the THRESHOLD macro in the othermalloc.h header file. If a request is too large for the initially selected section, the program will try searching the other chunk for the correct amount of space. If the right amount of space still cannot be found, unfortunately the program will not be able to accept the request, and will prompt the user with an error message.

Free will only free pointers that were allocated by the accompanying implementation of malloc. It will provide the user with errors whenever an attempt is made to use it in any other way, for example freeing a pointer that was not allocated, freeing a pointer multiple times, etc.

There are tests found in test.c, and can be run by calling "make all", running test with "./test" and then entering the number of the test case you would like to run.