**Analysis**

In my program, I have successfully implemented the init\_allocator function to initialize the memory necessary for the ackerman function to run. My program will provide users the capacity to allocation an amount of memory by using the command line:

**memtest -b*<block size>* -s*<memory size>***

Note: there is no space between –b and <block size>, so as –s and <memory size>

The running time of init\_allocator function will be O(n) with n is the is the number of basic block because the free list is created based on this number in a while loop. As the number of memory size increase, the running time will be decreased.

I also complete the my\_malloc function which will return the free memory block in the free list to the users. This function includes a Split function which will be called recursively to split the bigger block of memory to smaller ones to meet user’s requests. I expect this function will have the running time of O(log2n) with n is the number of time it is called. The running time also depends on the current free memory blocks available in the free list.

The my\_free function, unfortunately , is not fully implemented. It likely will cause segment faults when using with Ackerman function which calls my\_free many times. For that reason, I am not be able to test the running time of it as well as Ackerman’s running time yet.