As we can notice in our current algorithm, the memory tends to break into large number of intervals.

For example:

A screenshot of a cell phone

Description automatically generated

FreeList has one big segment of memory, but with an allocation of size 8192, memory breaks into five parts.

A close up of text on a black background

Description automatically generated

Our current algorithm suffers from significate time complexity due to large number of fragmentations, because the iterator has to spend more time to iterate through the list.

I would suggest to manage the freed and occupied memory blocks through any sorting algorithms, such as red-black tree, and sort the memory blocks using O(N log N) time.

By sorting the memory blocks, we could possibly increase the utility of the memory. My code is aborted whenever the requested memory cannot be found in the free list.