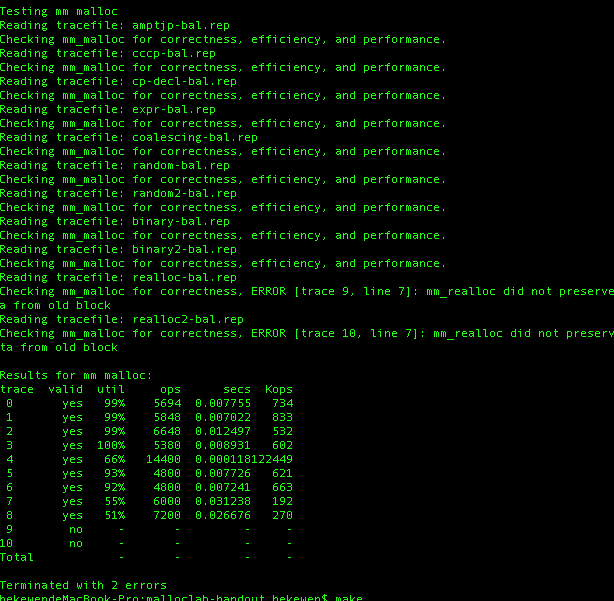
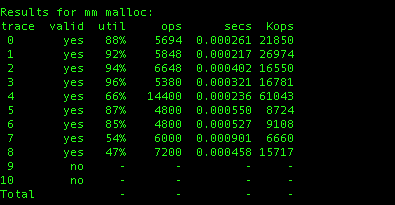
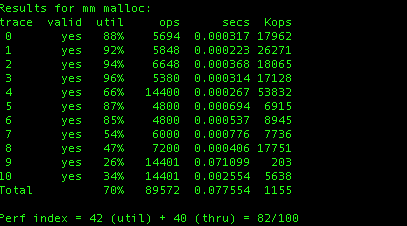
简单实现了一下realloc

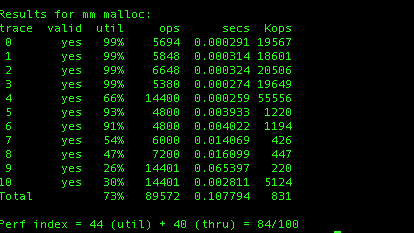
第一版：书上的简单隐式链表的实现

第二版：利用双向链表将搜索的时间复杂度降到heap中所有空闲堆的线性时间的显式内存分配器，时间有了大幅度的下降（直接从四位数的时间降低到了三位数），但是以牺牲空间利用率（由于空闲列表需要8个字节来存储前后指针，所以产生的小的外部碎片更多了）为代价。



加上realloc的时间，得分如下：  


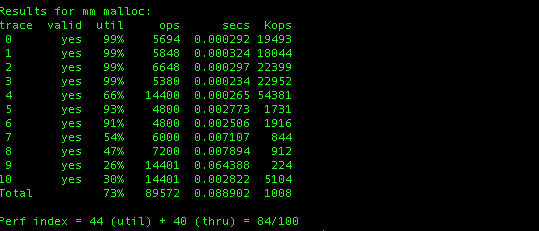
3. 进行地址排序空闲链表，可以有效的提高空间利用率

具体得分如下：  


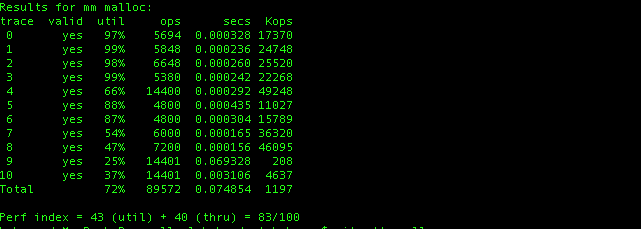
下面尝试加上尾指针：

进行初步的二分检索，可以选择从头还是尾开始检索。

得到的结果如下：遇到一些特定的case，效率还是提升了，有的甚至提升了快一半的速度，但是总的分数并没有提升。



4. 利用分离适配，速度提升提升不少，但是空间利用率却没有提升很多，由于即使速度提升了，但是速度的总分没有提升，一直都是40分，所以总的分数竟然下降了，这里考虑用最佳适配，提升空间利用率（T.T 这实验有BUG）



5.分离适配之后，将首次适配改进成最佳适配，牺牲了不少的效率。但是，提升了2%的吞吐量，万恶的coalecsing case .Fucking!!!!

