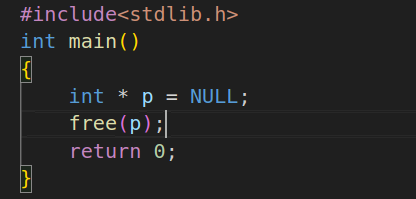
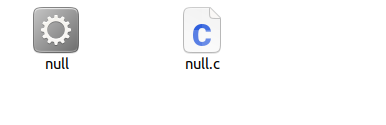
作业三

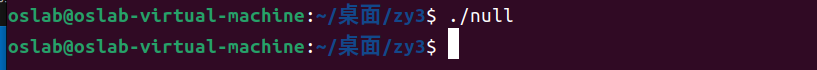
1.

设计程序的代码部分：



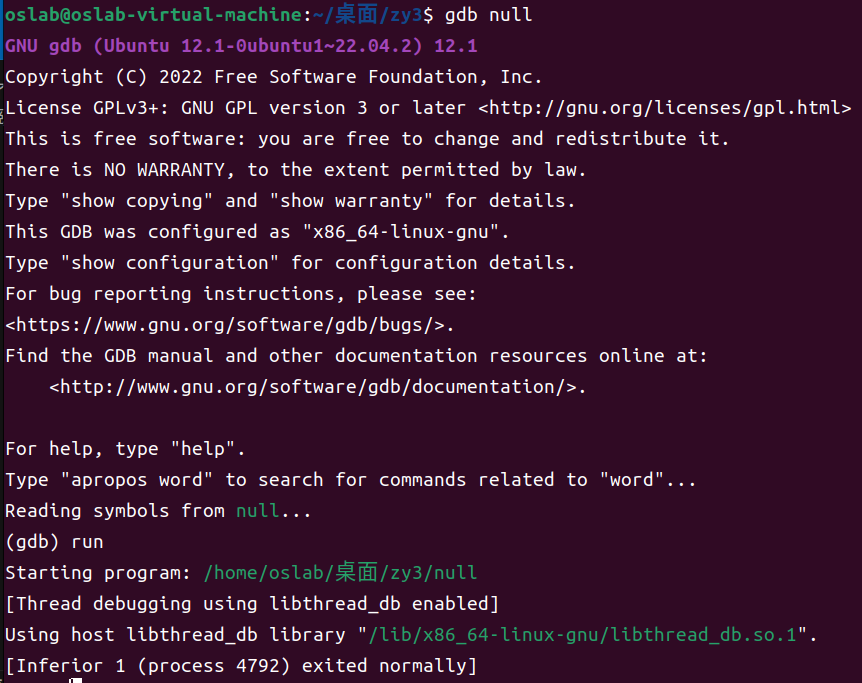
我们对其编译运行：





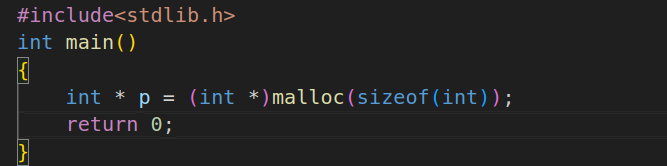
发现能正常编译运行，不会报错。

2.

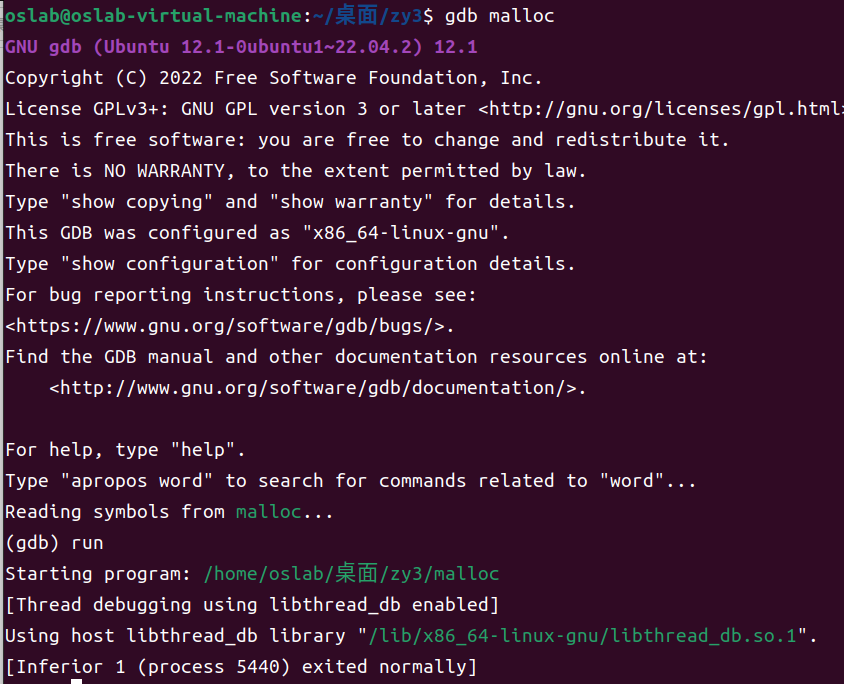
可以看到显示了Using host libthread\_db library "/lib/x86\_64-linux-gnu/libthread\_db.so.1".表示其加载了 libthread\_db 这个线程调试支持库。

4.

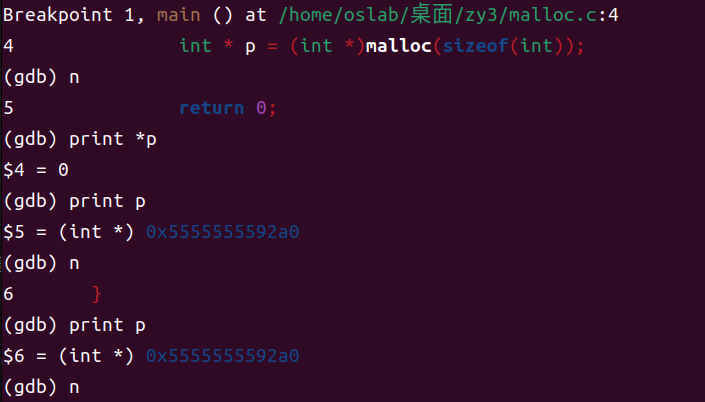
程序代码部分：



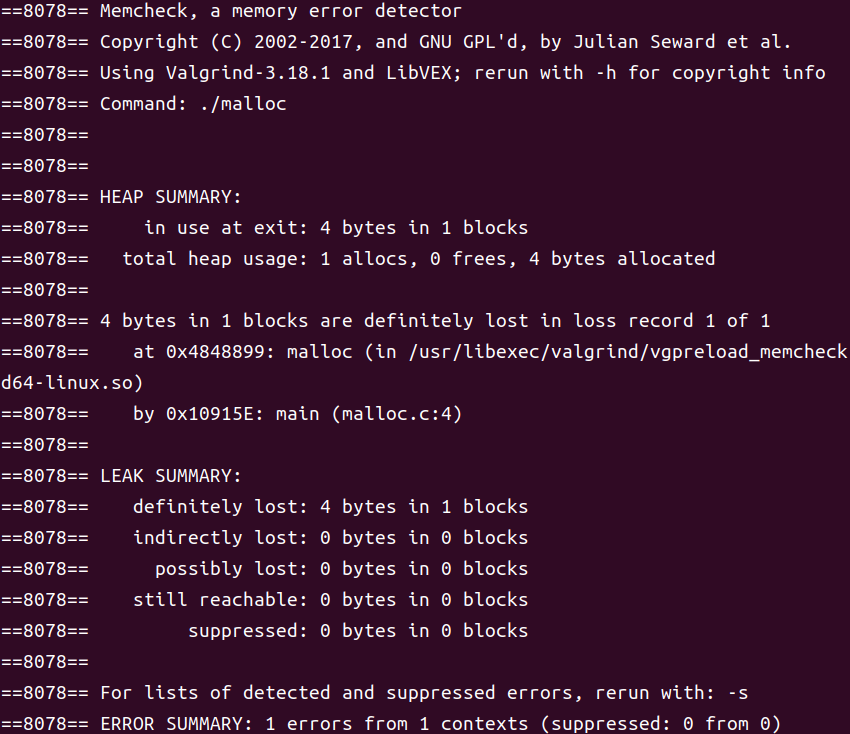
运行该程序：



发现并没有报错出现，但是程序本身具有内存泄漏的问题，所以我们使用gdb调试来查看该问题。在mian处设置断点，然后往下执行的时候查看其指向内存地址：



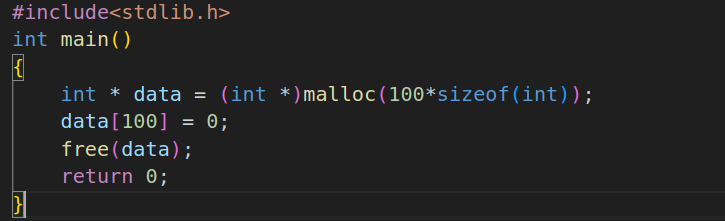
我们发现gdb调试并不能显示出内存泄漏的问题，我们需要使用Valgrind。



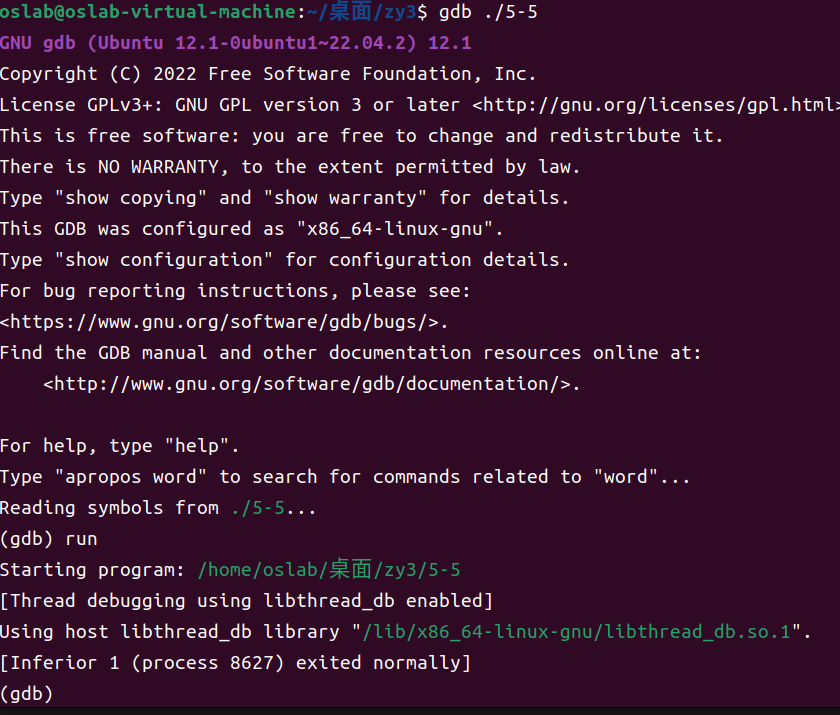
从结果中我们可以看到，程序在退出时仍有 4 个字节的内存未释放（很可能就是 malloc(sizeof(int)) 分配的那部分）。“definitely lost” 部分就说明存在明确的内存泄漏。

5.

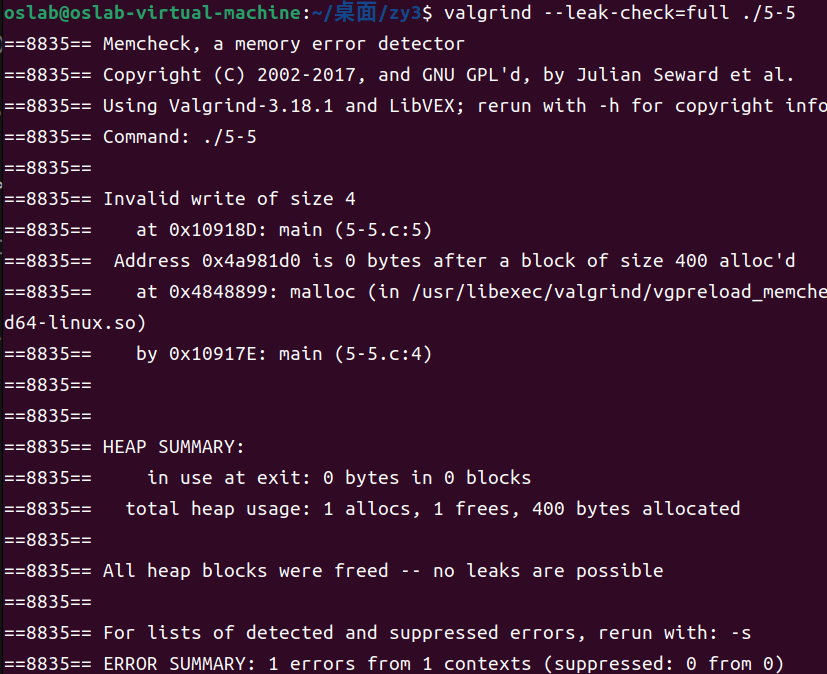
程序代码部分：



运行结果：



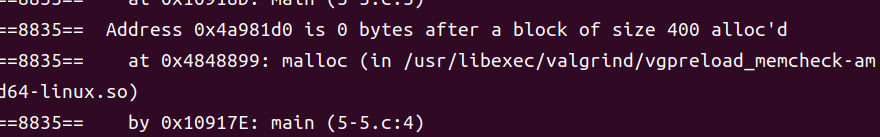
可以正常运行没有报错。再使用valgrind工具运行：



我们可以看到，其中有：



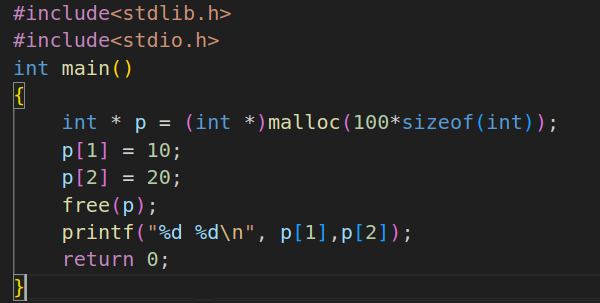
说明文件的第五行也就是内存分配有问题。



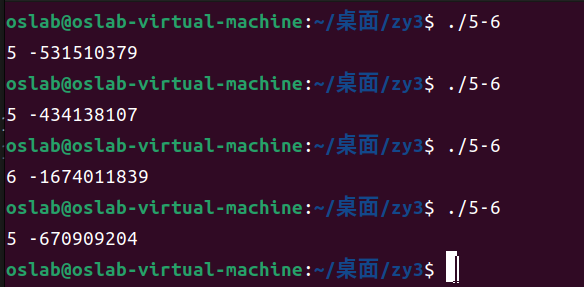
说明了存在越界写入的问题。

6.

程序代码部分：



运行结果：



看到每次运行之后打印出的值都不一样。