

实验二 openEuler 内存管理-页面淘汰算法

一、 实验说明

实验二与后续发布的实验三都包括两部分：必做+选做，每部分各占一半分值，所有同学只需要完整一个完整的实验（必做+选做）或两个实验的必做部分，总分即达到要求。即完成总分有三种基本选择：

- 1、 实验二必做+实验二选做
- 2、 实验三必做+实验三选做
- 3、 实验二必做+实验三必做

二、 实验背景

请求分页系统的基本思想是对每一个运行作业，只装入当前运行需要的一部分页面集合，即“工作集”，如主存资源紧张，可在原先装入主存的页面中选择一个或多个页，将其换出到辅存中，再把需要的页调入主存。

请求分页系统中当发现需要的某页内容未被调入主存，就要通过缺页中断处理程序调入该页。如这时主存中还有空闲的页架，那么只需要分配给调入页即可；如果此时主存中所有页架都已分配出去，就只能从已占用的页架中挑选一个页面，将其淘汰，腾出空页架以装入新页。选择需要淘汰的页有不同的算法，包括最优淘汰算法（OPT）、先进先出淘汰算法（FIFO）、最近最少使用淘汰算法（LRU）、最近未使用淘汰算法（NUR）等。

三、 实验目的

- 1、 以 openEuler 为例熟悉 Linux 的内存管理代码；
- 2、 掌握 LRU 的实现与拓展；

四、 实验要求

1、 必做部分

- 结合自己之前替换的 openEuler 内核版本源代码，（以 4.19.90-2209.5.0 为例），页面淘汰的源代码在 kernel-4.19.90-2209.5.0/mm/swap.c 及其他相关文件中。浏览完整代码并查阅相关资料，写一份报告，包括代码中所实现的算法流程图、重要的变量、数据结构、调用和依赖关系图（**必画**）等。
- 修改源代码并重新编译内核，实现功能：记录每次发生页面淘汰时淘汰的页面。报告中需写清修改的代码部分并进行解释，修改过的代码文件需写好注释，实验结果等。

2、 选做部分

修改源代码，在 LRU 之外，实现其他的页面淘汰算法之一。实现时可以不直接替换 LRU，但是需要添加数据结构，在每次页面淘汰时，记录按照自己实现的算法，会淘汰哪个页面。

五、 提交内容

最终提交的内容（根据自己的选题情况）可能包括：

- 1、 必做部分的报告（两个部分，重点是对过程的理解，**关系图、流程图要求亲自使用合适的画图软件制作**），必做部分修改的代码；
- 2、 选做部分的报告，选做部分修改的代码。

六、 提交时间

2024 年 5 月 26 日（十四周周日）23:59 之前将文件提交至 CANVAS。

注：第二次实验与第三次实验将合并提交，截止时间相同。提交时写清自

己的选题情况。