

# An implementation of page replacement algorithm

## Introduction:

Suppose each page can stores 10 commands and 4 memory blocks is assigned to a task. This application stimulates an overall page replacement process with a task calling 320 commands. For a task, there are two ways to call a command: in a random way or in a distributed way. As for the page replacement algorithm, the FIFO (First in First out) and LRU (least recently used) are implemented.

## Implementation:

- 1) FIFO: An index is used to keep track of all the pages in memory. When a page needs to be replaced, the page pointed by index will be selected.
- 2) LRU: A counter will be assigned to keep track of page usage over a short period of time for each memory. If a page kept in memory block is called, the counter of this memory block will be set to zero and all the other counter will plus 1. When calling a page out of the memory, the memory with a biggest counter will be selected to do a page replacement

## Result:

沈忆琛 103013

### 操作系统

——请求调页存储管理方式模拟

指令总数: 320  
每页指令: 10  
作业内存块: 4

index	command	kept page	loss page?	page out	page in
序号	指令	所在页	缺页	换出页	换入页
33	136	13	是	15	13
34	137	13	否	无	无
35	110	11	是	14	11
36	111	11	否	无	无
37	135	13	否	无	无
38	138	13	否	无	无
39	57	5	是	22	5
40	58	5	否	无	无
41	247	24	是	12	24

memory block

- page13
- page5
- page24
- page11

FIFO LRU random way distributed way 自动 automatic 手动 manual 清空 clear