LRU and FIFO scheduling algorithm

2014213404

19班

刘宇

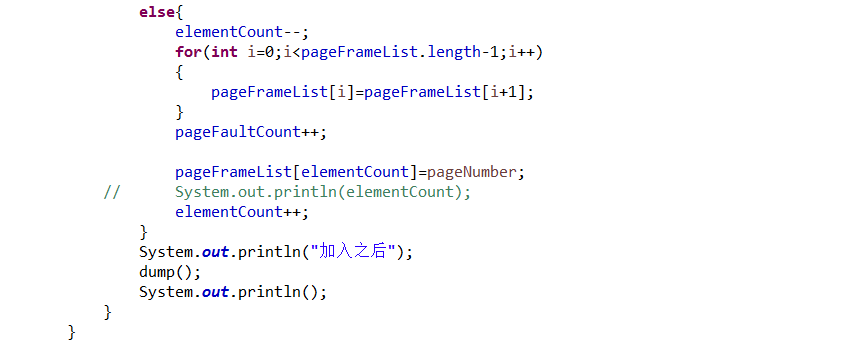
In the project, I achieve the insert() method in LRU.java and FIFO.java. Because LRU has different method from FIFO, I achieve them separately.

FIFO means first in first out. If there is no space in memory and the process needs a new page, the page should be brought to memory and another page which is the oddest page in the memory will be removed from the main memory. In the JAVA file, I use the array called pageFrameList to store all pages in the memory. And I place the latest page as the last element of the array. If the array has no space and there is a new page coming into the array, the first element of the array should be removed.



As for LRU algorithm, if a process needs a page which already exists in the memory, the page becomes the latest page and the sequence of pages in memory will be adjusted in order to reflect which page has not been used for a longer time. In LRU.java, I use the array called pageFrameList to store all pages in the memory. And the latest page is placed in the end of array. If the array has no space and there is a new page coming into the array, the first element of the array should be removed. If there is a page which already exists in array and be used by process again, the page should be placed in the end of array.





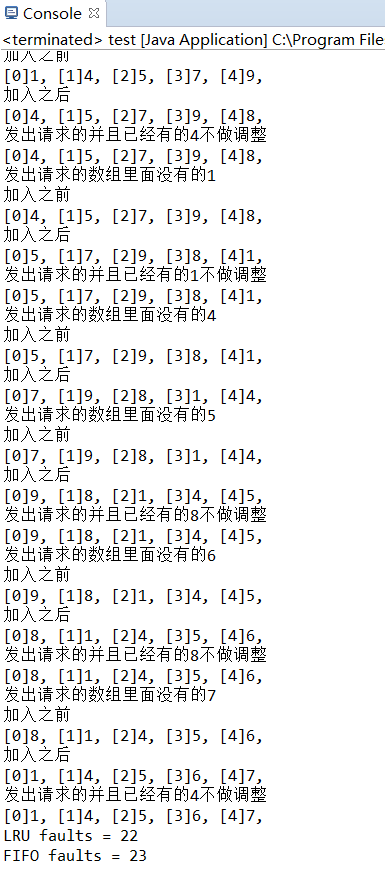
Analyze the result

I set the range of pageNumber is from 0 to 9.

I use different reference string size and number of frame to compare two algorithms.

1. reference string size: 50

number of frame: 5

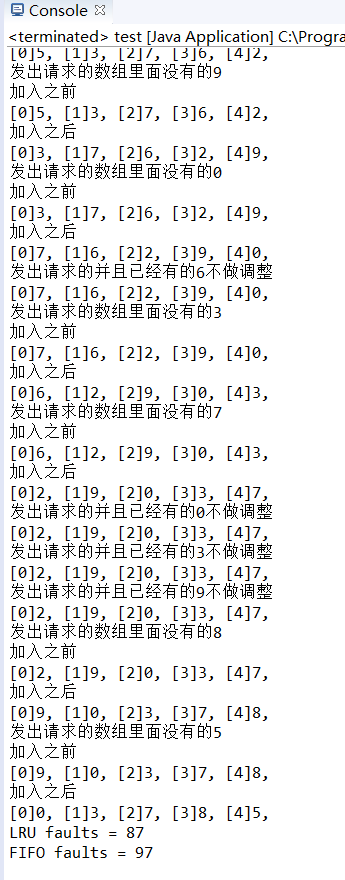


I run the program many times. And the results change many times.

I find that the number of LRU faults is approximate to the number of FIFO faults. They almost have same performance.

1. reference string size: 200

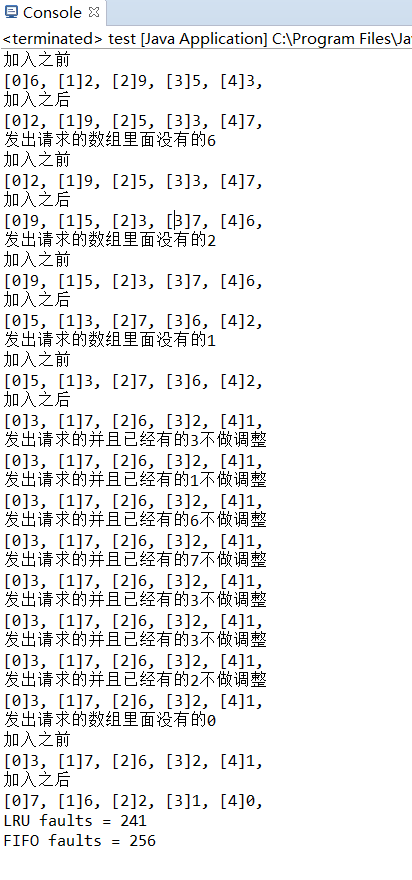
number of frame: 5



The number of LRU faults is normally less than number of FIFO faults. However, the difference between the number of LRU faults and the number of FIFO faults is not large.

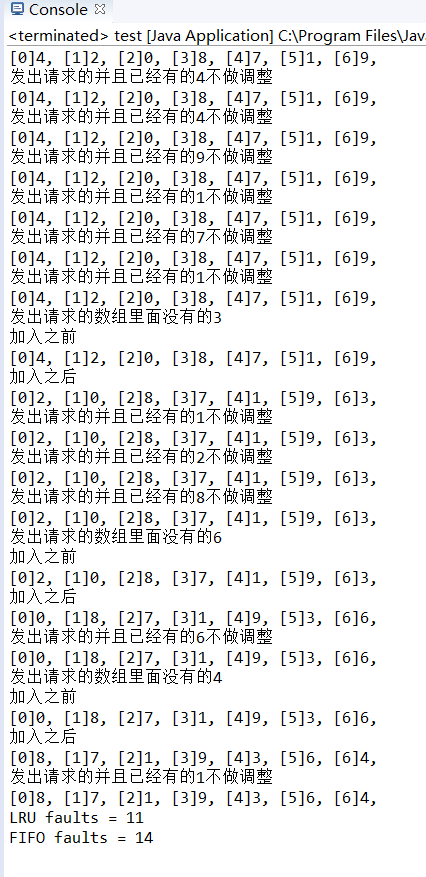
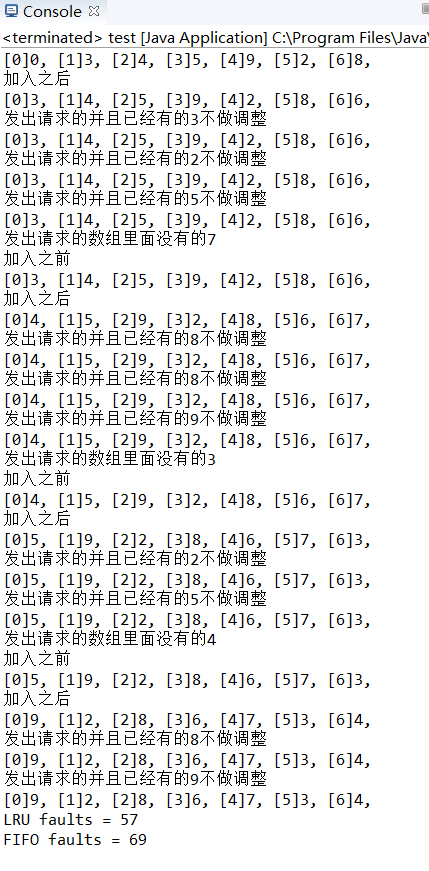
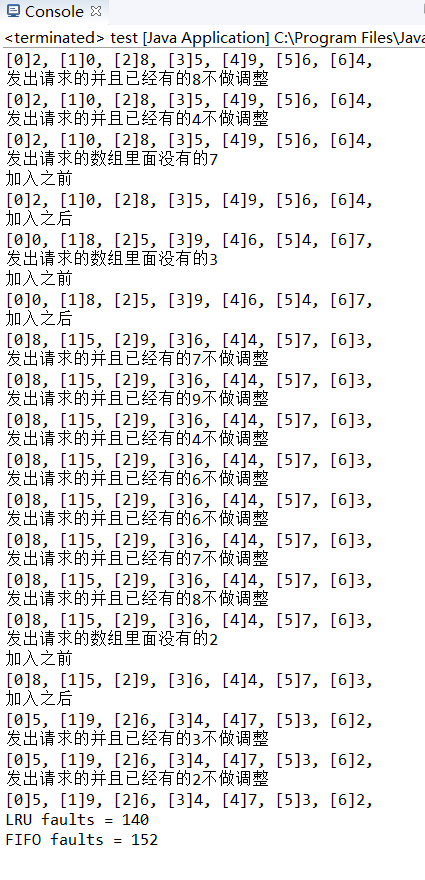
1. reference string size: 500

number of frame: 5



The LRU has fewer faults than FIFO when the number of reference string size is larger.

If number of frame is 7 and reference string size is 50,200,500

So by setting different values of reference string size, I find that the LRU has fewer faults than FIFO so LRU has better performance.