

### **Title: LRU (Least Recently Used) page replacement algorithm**

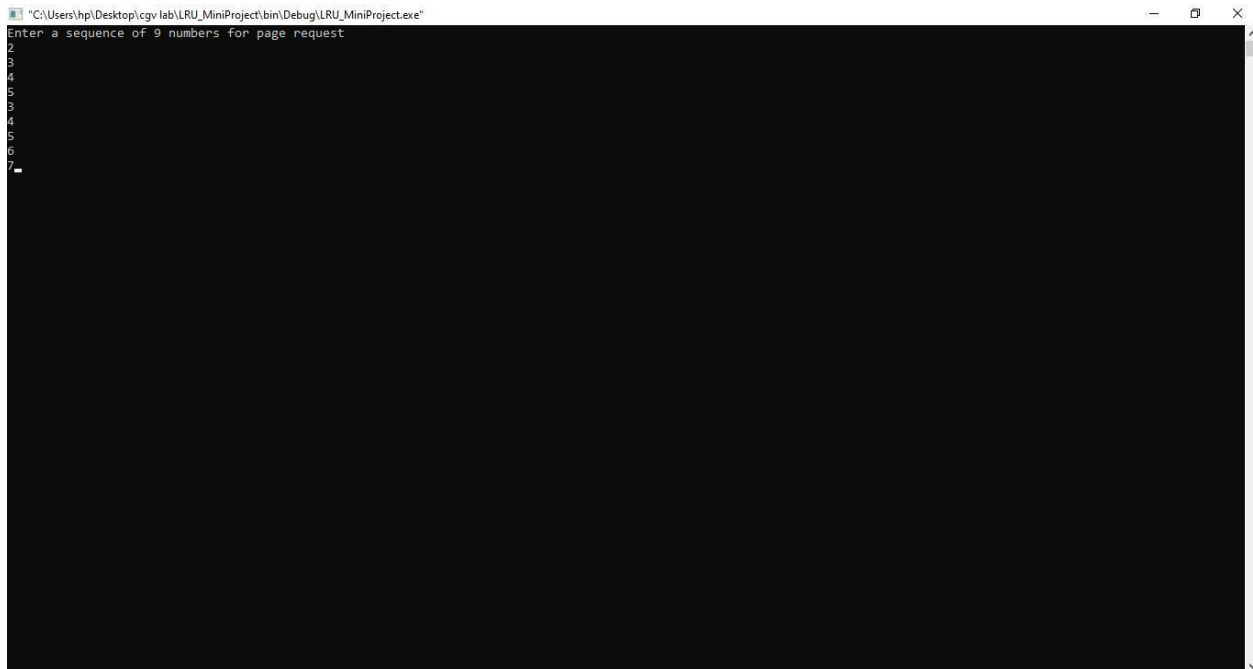
#### **Aim of the Project:**

A good approximation to the optimal algorithm is based on the observation that pages that have been heavily used in the last few instructions will probably be heavily used again in the next few. Conversely, pages that have not been used for ages will probably remain unused for a long time. Therefore the aim is, when a page fault occurs, throw out the page that has been unused for the longest time.

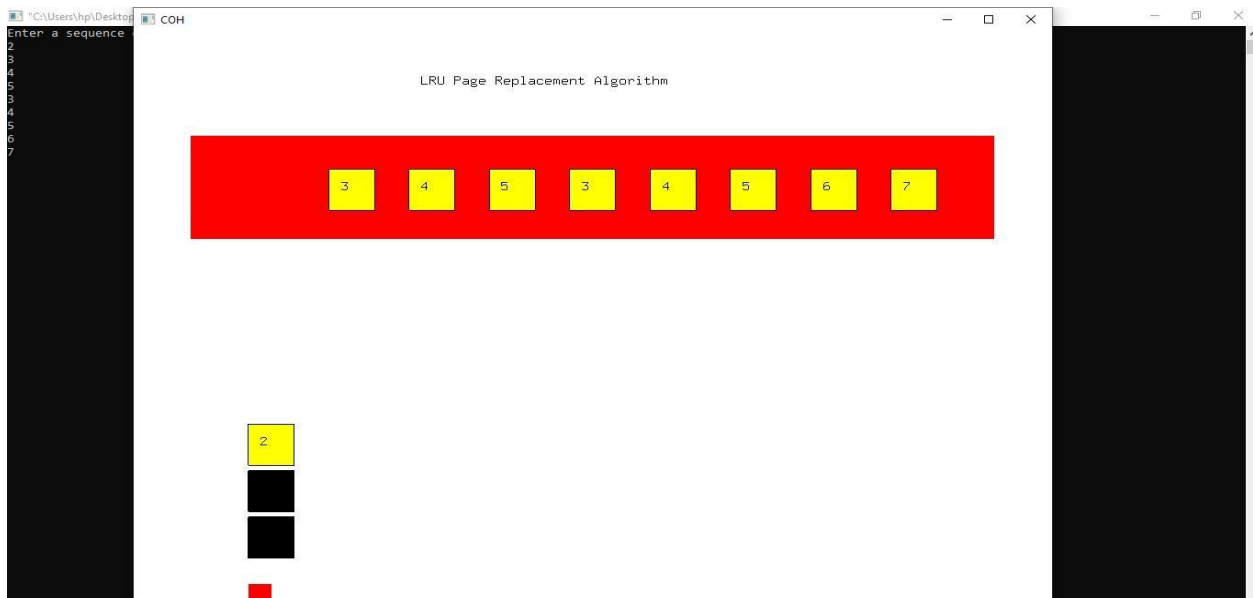
#### **Objectives:**

- Locality Principle
  - A block most recently used (MRU Block) could be accessed again soon. On a cache miss, block determined to be evicted should be least-recently used block (LRU block).
  
- LRU Design
  - Each way in a set is attached to a counter to record the accesses recently.
  - This enables to identify MRU & LRU block. When set is full, LRU block is replaced.
  
- LRU-Like Design
  - Based on Augmented Cache Design, buffer used to filter the LRU Blocks.
  - Holding all MRU Blocks in DM Cache in increase the hit rate.

## Sample output

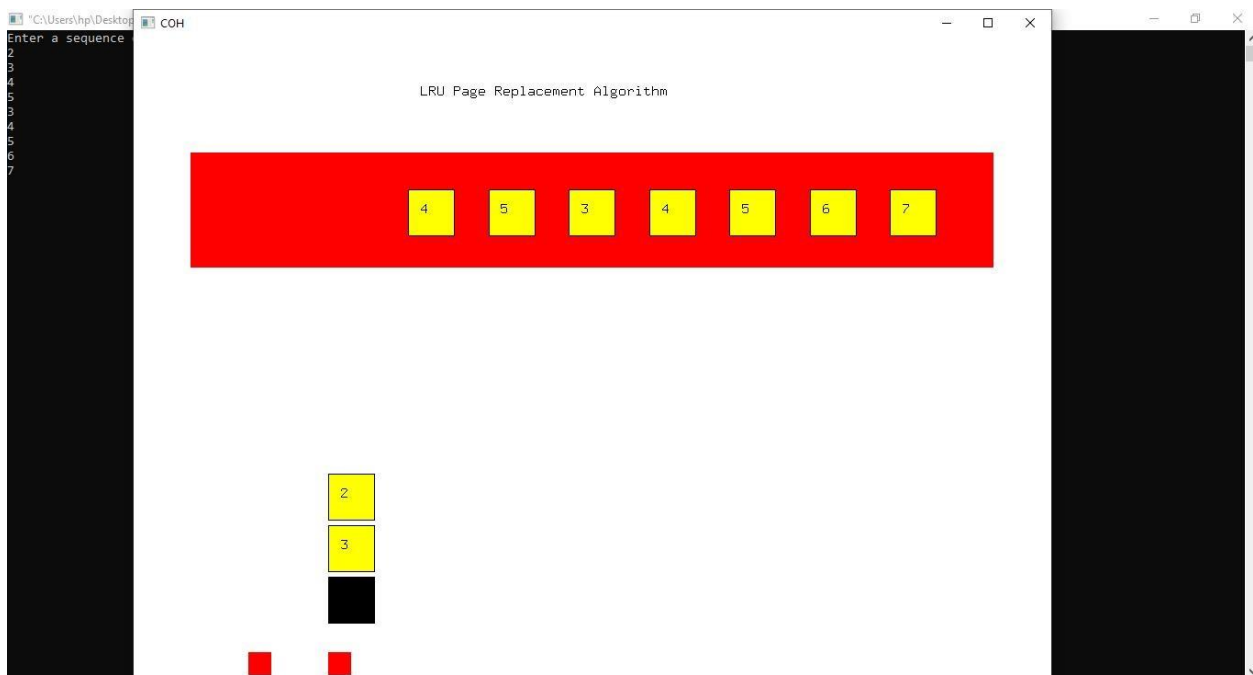


### 5.1 LRU Home Screen

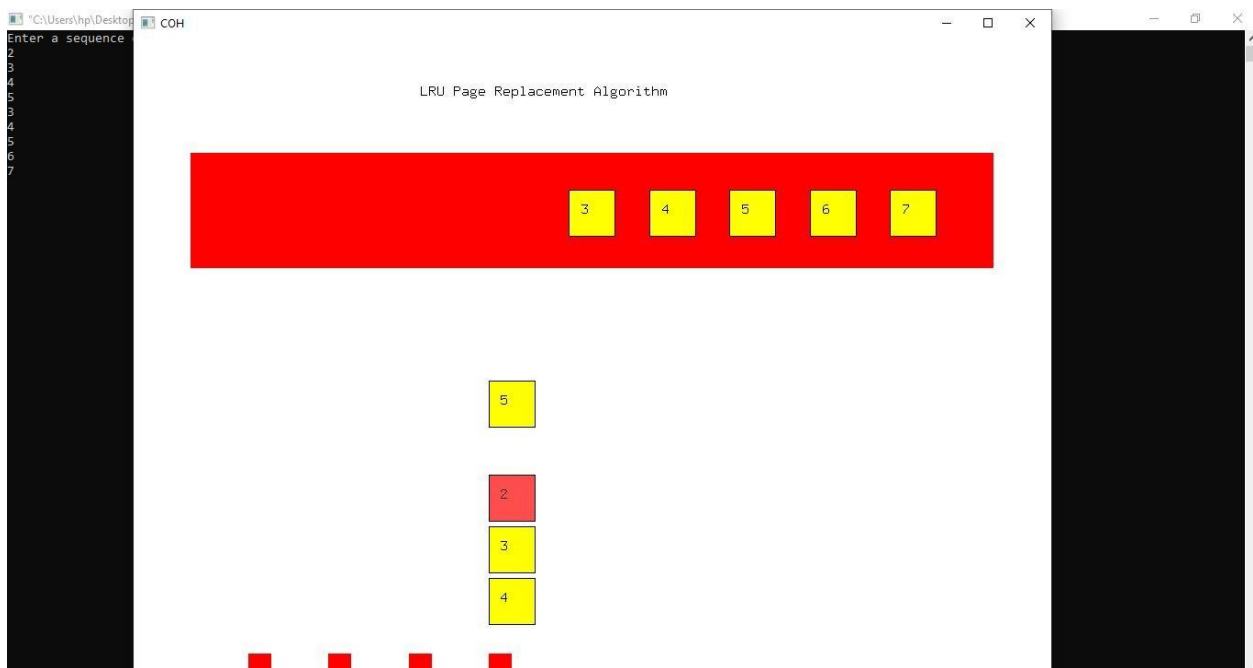


**5.2 LRU with one page. The request was for two and two was not present in the page frame so it is placed in the first page.**

## Simulation of LRU using OpenGL

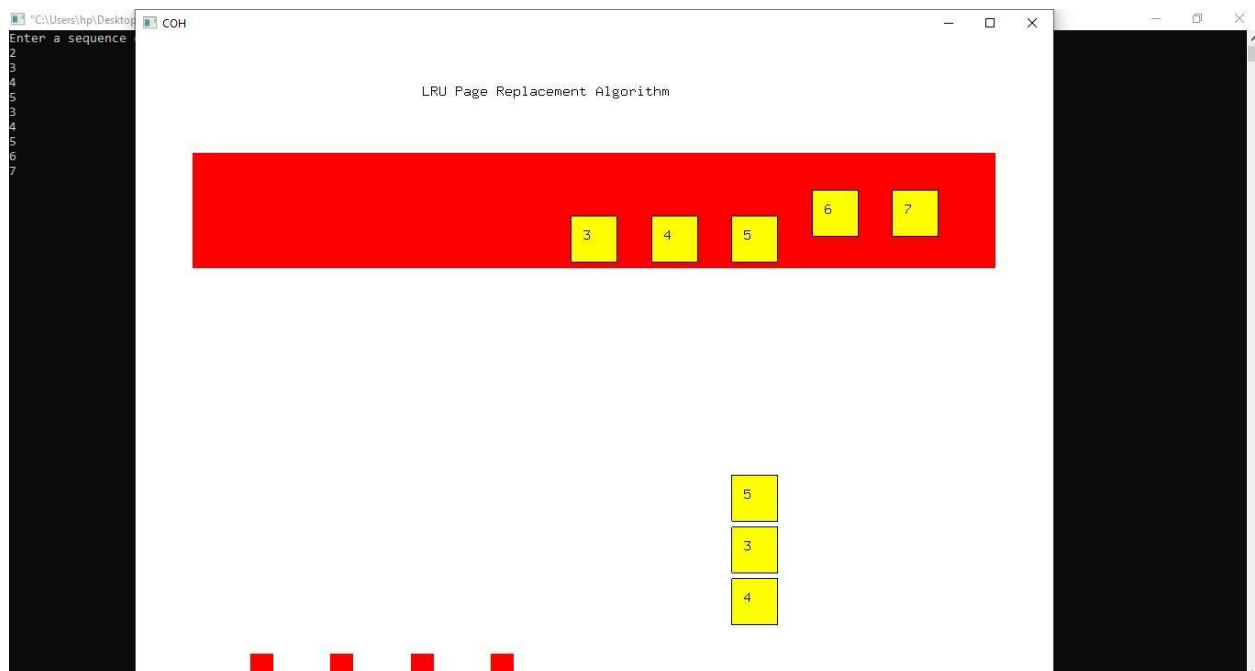


**5.3 LRU with two pages. The request was for three and three was not present in the page frame so page fault occurred and it is placed here.**

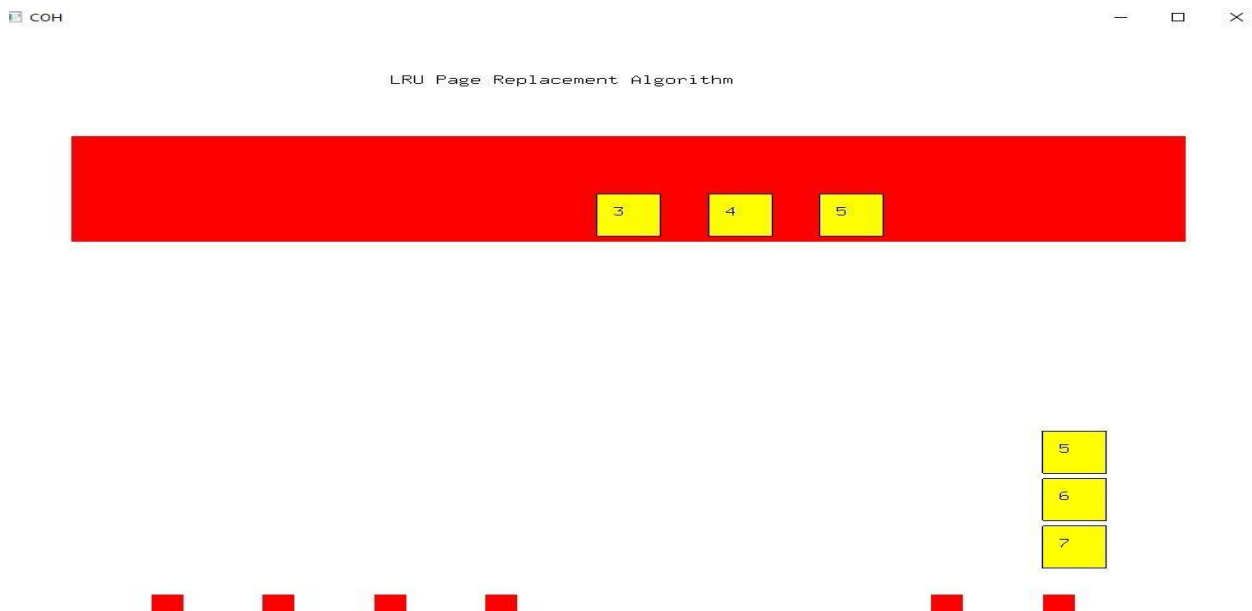


**5.4 The request is for five. The least recently used page that is two will be replaced with five.**

## Simulation of LRU using OpenGL



**5.5** Five, three and four are placed without any page fault.



**5.6** LRU Last Screen