Page Replacement Algorithm

Group - 29

Anuj Shah –1401084
Charvik Patel –1401079
Himanshu Budhia –1401039
Maharsh Patel –1401109

School of Enginering and Applied Sciences - Ahmedabad University

December 8, 2016

Outline

Introduction

- Paging is a memory management scheme by which a computer stores and retrieves data from secondary storage for use in main memory.
- In Paging, the operating system retrieves data from secondary storage in same-size blocks called pages.

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Why Page Replacement ?

- Program may be occupy Larger space in main memory.
- Operating system brings a few pieces of the program into main memory for execution ,thus page replacement is required.
- To Reduce the swapping time Different Algorithm is required.
- Thrashing

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Outline

- Introduction
- Why Page Replacement ?
- Algorithm
  1. FIFO
  2. LFU
  3. LRU Counter
  4. LRU Stack
  5. MFU
  6. OPT
  7. Second Chance
- Test Result
- Technical Specification
- References

Technical Specification

- Code Language – C and Python 2.7 .
- Code compatibility – UNIX and Windows.
- Input – Randomly generated page numbers using Python.
- Output – Number of page faults and intermediate pages in frame.
- Data structures used – Stack, Array, Linked list.

Outline

References

- Operating System internals and Design Principles, 7th ed. Pearson
- "LRU Implementations", Cs.jhu.edu, 2016. [Online]. Available: http://www.cs.jhu.edu/~yairamir/cs418/os6/tsld021.htm. [*Accessed* : 07 – *Dec* – 2016].
- "Operating System - Virtual Memory", www.tutorialspoint.com, 2016. [*Online*]. Available: https://www.tutorialspoint.com/operating_system/os_virtual_memory.htm. [*Accessed* : 07 – *Dec* – 2016].
- "Operating Systems", Www2.cs.uregina.ca, 2016. [*Online*]. Available: http://www2.cs.uregina.ca/~hamilton/courses/330/notes/memory/page_replacement.html. [*Accessed* : 07 – *Dec* – 2016].
- "Page replacement algorithm", En.wikipedia.org, 2016. [*Online*]. Available: https://en.wikipedia.org/wiki/