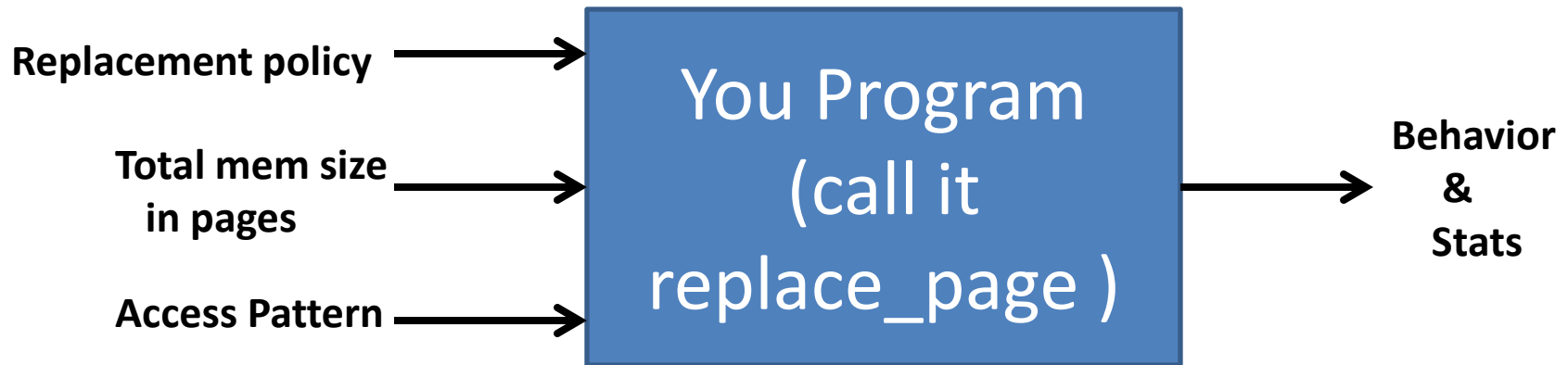


Page Replacement Policy

The Aim Of This Project

Implement several replacement policies and compare between them in terms of performance.



Replacement Policy

0: FIFO

1: Second chance

2: LRU

Total Memory Size in Page

- A positive non-zero number
- Indicates the total number of pages that can exist in memory at the same time

Access Pattern

- A text file
- Each line contains a page number, means that this page is accessed
- The file indicates pages access over time.
- Page number ≥ 0
- Example:

7

1

9

Behavior and Stats

- A text file
- The file is divided into two parts:
 - Memory map
 - Statistics
- The memory map prints in a single line the numbers of the pages available in memory after each access
- Stats:
 - Percentage of page faults ($\frac{\text{\#page faults}}{\text{\# accesses}}$)

Example

replace_page 0 3 access.txt

Means: apply FIFO to a memory that can hold up to 3 pages and the accesses are in file

access.txt

access.txt

7

8

9

7

4

5

The output filename is the input file name with extension dependent on replacement policy, followed by .txt
Here for example: access.fifo.txt

Example: Output

7

7 8

7 8 9

7 8 9

4 8 9

4 5 9

7
8
9
7
4
5

Percentage of Page faults = 0.83

What To Submit

- Readme file, contains the following info:
 - How to compile your code
 - e.g. `gcc -o name file1.c file2.c`
- Source code
- gzip the whole files in a one file:
yourlastname.zip
- Email it to your assigned grader

Avoid The Following Mistakes (Penalty applied for each)

- TA must **not** change your source code
- Code **does not run on CIMS** machines
- You used a tool to compile your code not present on CIMS machines
- You use a language other than C/C++/Java
- Late submission (-1 for each day)