

Nirbhay Sharma (B19CSE114)

Features:

All the data structure such as Page table, TLB, Free frame list are implemented

Page fault handling is done using the LRU algorithm

The maximum size of the TLB is taken care of in case maximum size in TLB is reached the

How to Run:

C++ is a prerequisite to run the code

Open terminal and type the following command: `g++ DemandPaging1.cpp ;./a.out`

Inputs required:

K(Number of process)

M(maximum number of pages per process)

F(frames in the main memory)

S(maximum size of TLB) (please note that it should be atleast 1)

Later things it will automatically generate randomly

Observations and Explanation:

By incorporating TLB in the whole process we get very fast lookups and our time for accessing the memory is saved

One disadvantage that can be overserved here is that if two process refer to the same page then TLB will not be able to get us the correct answer so one process should complete its execution and then another process should do a fresh start by flushing the content of TLB otherwise it will not work well

Limitations:

If maximum size of TLB is too low then the lookups in TLB will not be very beneficial as at a particular time very few entries are there in TLB so more chances of getting TLB miss are there

Input for Results.txt file

The following input is given for generating the results.txt file -> 4 8 5 2 (changing the input will also change the content of the result.txt file)