CS570 Spring 2014 Assignment 2

This page last modified 11 March, 2014

You shall implement and analyze Page Replacement Algorithms for OS Virtual Memory Management

1. You shall create a program which implements the following Page Replacement Algorithms:
    - OPT (Optimal)
    - LRU
    - FIFO
    - Clock

    This program shall simulate a process accessing its pages during execution. The sequence of pages it accesses will be contained in a file, named "pages.txt" which you shall create. This file shall be located in the same directory as the executable and shall contain a series of positive integers representing frame numbers (simulating part of a virtual address) of frames being referenced when a process is running. These integers shall be in the range of 1..99, each one will be separated by a space character. Note, only one string of integers will exist in this file.

2. The program shall perform the following:
    - Upon startup, prompt the user for number of frames of main memory the system will have
    - read the sequence of pages accessed from the file "pages.txt"
    - Run each of the algorithms (OPT, LRU, FIFO, and Clock) on the same input string from the step above
    - Printout the number of faults each algorithm generates to the terminal.

    In your README file (your README file may be a plain text file or MS Word formatted file), in addition to the required information (see READMEformat.pdf file posted on BB), include a summary of your findings from your analysis. Run your algorithms on at least two different input data sets and at least two different numbers of frames of memory, then compare the performance of the algorithms (include a conclusion and defense of your conclusion). Use of spreadsheets, charts, or graphs to support your explanations is very much encouraged.

    I will test your program by compiling it and executing it on rohan. Your program shall be written such that it compiles and executes cleanly when using any compiler/interpreter available on rohan. Note - you must use a Makefile.

    You shall create a sub-directory named "a2" in your home directory. In it, you shall place all of your project files, including your Makefile and a README file. Your source files

SHALL CONTAIN sufficient comments for making the source easy to read. Points will be taken off for poorly (or non) commented source. Name the executable "a2".

- Create ~/a2 by hand.
- Create all necessary project files. Put them into ~/a2.
- The Makefile shall create an executable named "a2" in this same directory (~/a2).
- The system call "system()" will NOT be allowed
- You are working individually or in teams of two this assignment.
- You may use the cc, CC, gcc, or g++ compiler on this assignment.

The assignment is due 1730 on Monday, 7 April 2014

TURNING IN YOUR WORK:
Follow the turn-in procedures posted in the course's Blackboard (as stated in the Assignment_Turning.pdf file).