

New York University

Tandon School of Engineering

Department of Electrical & Computer Engineering

Introduction to Operating Systems (CS-GY6233)
Fall 2021

Assignment 8
(10 points)

Use C, C++, python or matlab to develop a program whose main routine accepts two parameters n and k , i.e. when you invoke your program from the shell, you pass it two parameters, n and k , where $n \geq 16$ and $k \geq 3$ and is in powers of 2 (e.g. 2, 4, 8, 16, etc.). Your main routine shall generate a random page trace of size n , where the page numbers have values ranging from 0 to $k - 1$.

Develop a subroutine within your program that implements the LRU page replacement algorithm (as a separate function within your program). Your algorithm shall use the doubly linked list stack implementation as outlined in slide 29 of lecture 10). The function shall accept a page trace and a parameter f for the number of frames allocated.

Your main routine shall then apply the random page trace to the subroutine implementing the page replacement algorithm, multiple times (using only one trace, randomly generated), passing a parameter f (number of page frames used) that ranges from 4 to k . Your main routine shall then record the number of page faults for each run (i.e. for each f).

Run your program using a page trace of length $n=64$, $k=16$ (thus f ranges from 4 to 16). Plot and submit a graph displaying the number of faults vs the number of frames allocated.

Can your outcome exhibit the Belady's anomaly?

Notes:

- For debugging, you may use the page trace given in the lectures instead of randomly generating it (with $n=20$, $k=8$ and $f=3$) and verify you obtained the same results as in the lectures.
- You may output into a text file (e.g. a comma separated file) and then use excel to plot.

Submission file structure:

Please submit a **single .zip file** named **[Your Netid]_lab#.zip**. It shall have the following structure (replace # with the actual assignment number):

```
└── [Your Netid] hw# (Single folder includes all your submissions)
    ├── lab#_1.c (Source code for problem 1)
    ├── lab#_2a.c (Source code for problem 2a, and so on)
    ├── lab#_1.h (Source code header file, if any)
    └── Makefile (makefile used to build your program, if any)
```

└─ lab#.pdf (images + Report/answers to short-answer questions)

What to hand in (using Brightspace):

- Source files (.c or .h) with appropriate comments.
- Your Makefile if any.
- A .pdf file named “**labX.pdf**”, containing:
 - Screen shot(s) of your terminal window showing the current directory, the command used to compile your program, the command used to run your program and the output of your program.

RULES:

- You shall **use kernel version 4.x.x or above**. You shall not use kernel version 3.x.x.
- You may consult with other students about GENERAL concepts or methods but copying code (or code fragments) or algorithms is NOT ALLOWED and is considered cheating (whether copied from other students, the internet or any other source).
- If you are having trouble, please ask your teaching assistant for help.
- You must submit your assignment prior to the deadline.