Republic of the Philippines
Tarlac State University
COLLEGE OF COMPUTER STUDIES
Tarlac City, Tarlac
Tel. No. (045) 6068173

Case Study in Operating Systems

Submitted by:
Milar, Rose Marie C.

Submitted to:
Mrs. Jo Anne G. Cura

May 21, 2025
A.Y 2024 – 2025

I. User Interface (UI)

1.1 Landing Page

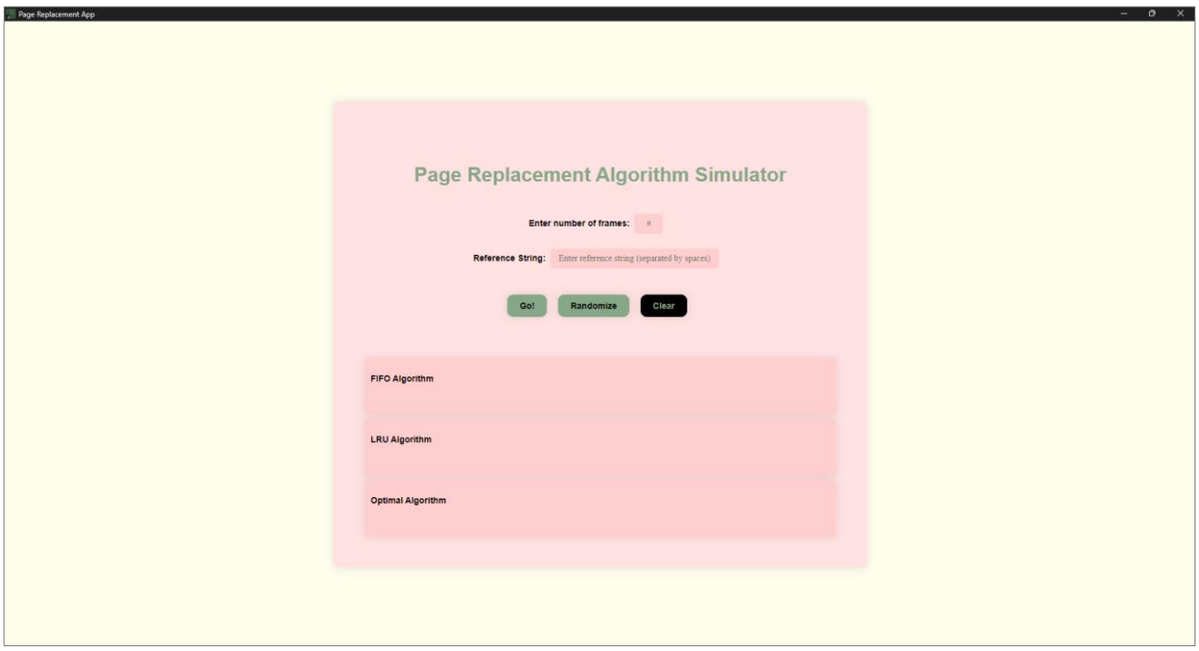

Figure 1. Landing page of the program

This figure shows the main screen of the Page Replacement Algorithm Simulator. Users can enter the number of page frames and a reference string (a list of page numbers). There are three buttons below the input fields: “Go!” to run the simulation, “Randomize” to generate a random reference string, and “Clear” to reset the inputs. The lower part of the page contains sections for showing the results of the FIFO, LRU, and Optimal algorithms after the simulation is run.

1.2 Icon

Figure 2. Icon of the program

This figure shows the official icon of the application, used for identification on the desktop or taskbar.

II. Action Buttons

2.1 Randomize Button

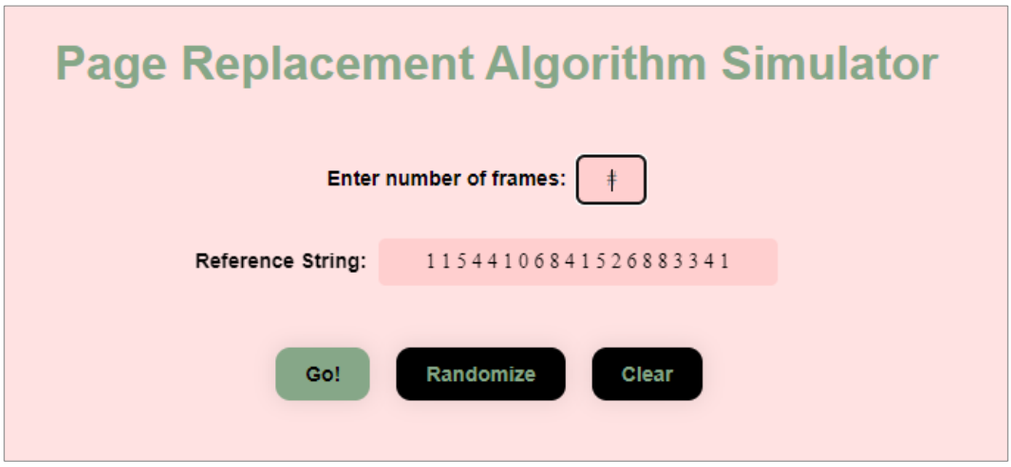

Figure 3. When randomize button is triggered

This figure illustrates the result of triggering the "Randomize" button, which generates a new page-reference string automatically.

2.2 Validation

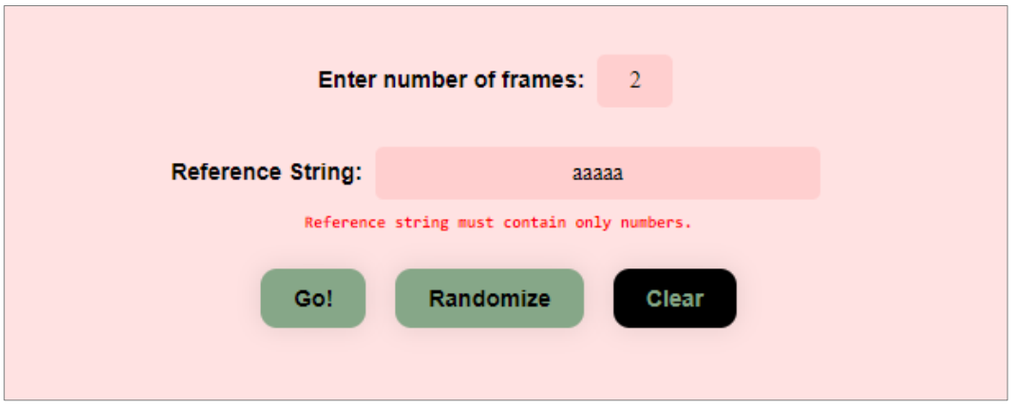

Figure 4. Validation for the inputs

This figure demonstrates the program's input validation feature, where users are notified when an invalid data is entered.

III. Outputs

3.1 Output 1

Frames: 4

Reference String: 1 1 5 4 4 1 0 6 8 4 1 5 2 6 8 8 3 3 4 1

Figure 5. Output of the program with 4 frames and a random reference string

This figure shows the result of running the page replacement algorithms using 4 frames and a reference string of 20-page numbers. The output includes how each algorithm (FIFO, LRU, and Optimal) handled the page references and the number of page faults that occurred.

3.2 Output 2

Frames: 3

Reference String: 0 0 1 1 3 9 6 9 2 4 1 1 1 2 6 7 0 1 0 4

Figure 6. Output of the program with 3 frames and a different reference string

This figure presents the simulation result using 3 frames and a new set of 20-page references. Each algorithm's performance is displayed, highlighting the steps and total page faults for comparison.

3.3 Output 3

Frames: 3

Reference String: 9 5 0 5 7 7 3 6 8 7 6 4 8 1 6 7 2 4 6 5 4

Figure 7. Output of the program with 3 frames and a longer reference string

This figure shows the algorithm outputs using 3 frames and a longer reference string of 22 numbers. The results include how each algorithm managed memory and the total page faults for each case.