Page Replacement Simulator

Prepared By:

Tomas G. Jose

BSCS-3B

Submitted to:

Jo Anne G. Cura

Instructor

May 21, 2025

Table of Contents

[Introduction 4](#_Toc198600960)

[Page Replacement Algorithms 5](#_Toc198600961)

[FIFO (First-In First-Out) 5](#_Toc198600962)

[LRU (Least Recently Used) 5](#_Toc198600963)

[Optimal 6](#_Toc198600964)

[The User Interface 6](#_Toc198600965)

[Configuration 7](#_Toc198600966)

[Reference String 7](#_Toc198600967)

[Simulation State 7](#_Toc198600968)

[Options 7](#_Toc198600969)

[Controls 8](#_Toc198600970)

[Status & Info 8](#_Toc198600971)

[Sample Outputs 9](#_Toc198600972)

[Simulation 1 [ 6, 3, 6, 9, 5, 5, 0, 5, 4, 2, 7, 0 ] 9](#_Toc198600973)

[FIFO Algorithm Outputs 9](#_Toc198600974)

[LRU Algorithm Outputs 19](#_Toc198600975)

[Optimal Algorithm Outputs 28](#_Toc198600976)

[Simulation 2 [ 5, 3, 2, 5, 9, 3, 8 ] 37](#_Toc198600977)

[FIFO Algorithm Outputs 37](#_Toc198600978)

[LRU Algorithm Outputs 43](#_Toc198600979)

[Optimal Algorithm Outputs 48](#_Toc198600980)

[Simulation 3 [ 8, 0, 1, 8, 3 ] 54](#_Toc198600981)

[FIFO Algorithm Outputs 54](#_Toc198600982)

[LRU Algorithm Outputs 59](#_Toc198600983)

[Optimal Algorithm Outputs 63](#_Toc198600984)

[Conclusion 68](#_Toc198600985)

# Introduction

In the field of operating systems, having an effective memory management is very crucial in optimizing system performance and ensures a smooth and efficient process execution. One key challenge in memory management is finding out which memory pages to replace whenever a page fault happens. In order to address this, different page replacement algorithms are developed, each with its own way on selecting pages to be removed from memory. Among them, the most common ones are FIFO (First-In First-Out), LRU (Least Recently Used), and Optimal Replacement.

This project presents a Page Replacement Algorithm Simulator developed with the use of PySide6, a Python framework for building graphical user interfaces. The simulator shows a visual and interactive way to understand how each algorithm works in managing memory frames during a sequence of page access. Users can input the length of the reference string (maximum of 12) and the total number of frames to be used. The application will then generate the reference string. Afterwards, users can select an algorithm, and observe the simulation step-by-step. The application will then show if it is a page hit or page miss, and track the total number of page faults for each simulation. By offering a user-friendly interface and real-time feedback, the simulator could serve as an educational tool for students and practitioners seeking to understand and compare page replacement algorithms.

# Page Replacement Algorithms

## FIFO (First-In First-Out)

As the name suggests, FIFO replaces the oldest page in memory—the one that was loaded first—whenever a new page needs to be brought in and there are no free frames available.

In this algorithm, pages are loaded into memory in the order they arrive, and each page is placed in a queue. When the memory is full and a new page must be added, the page at the front of the queue (i.e., the page that has been in memory the longest) is removed, and the new page is added to the rear of the queue. This approach does not take into account how frequently or recently a page has been accessed; it operates purely based on arrival time.

## LRU (Least Recently Used)

In LRU, when a new page must be loaded into memory and there are no free frames, the page that has not been accessed for the longest time is selected for replacement.

LRU keeps track of the usage history of pages, either by timestamps or by maintaining a stack or list that is updated with each access. When a page is accessed, it is marked as the most recently used. If a page fault occurs and memory is full, the least recently used page is identified and replaced.

## Optimal

The Optimal page replacement algorithm provides the theoretical best performance in terms of minimizing page faults. It works by replacing the page that will not be used for the longest period of time in the future. Unlike FIFO or LRU, the Optimal algorithm looks ahead in the reference string to determine which page can be safely removed without being needed soon.

When a page fault occurs and memory is full, the algorithm examines the remaining reference string and selects the page that will be accessed farthest in the future—or not at all—for replacement. This ensures the least number of page faults for any given reference string and number of frames.

# The User Interface

The application has six different sections, each having their own different purpose in the simulation and analysis of page replacement algorithms. The full user interface layout is shown in Figure 1.

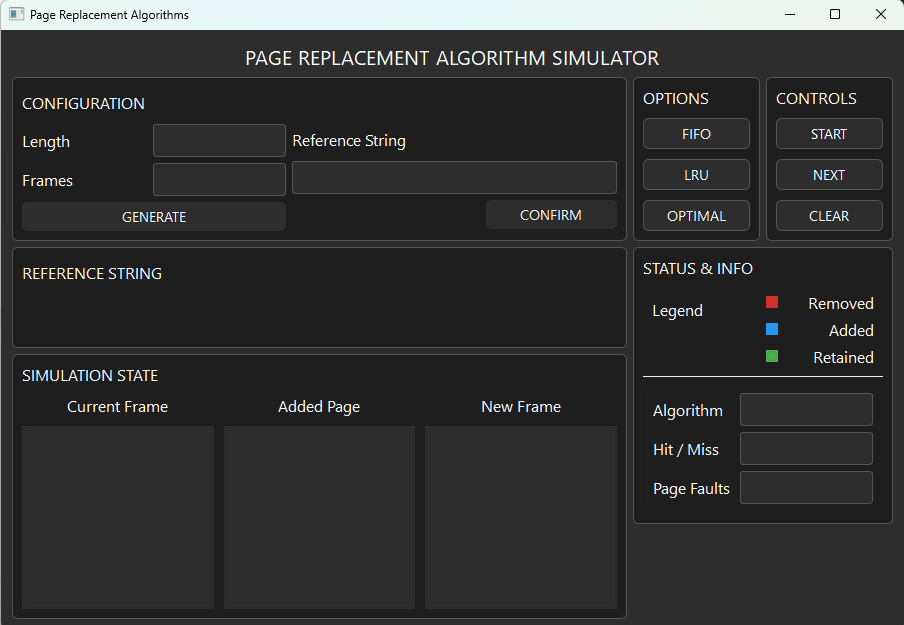


Figure 1: User Interface

## Configuration

The purpose of this section is to allow users to configure the simulation settings. It includes input fields for the length of the reference string with the maximum accepted length as 12 and the number of memory frames to be used. Users can click the Generate button to create a random reference string based on the specified length, and will be displayed at the adjacent box. Once satisfied, the user can press Confirm to finalize the reference string which will be used in the simulation

## Reference String

This section displays the complete reference string generated. It provides a visual list of page numbers that will be accessed during the simulation, allowing users to anticipate how each algorithm will respond to the upcoming page requests.

## Simulation State

The simulation state area visually presents the behavior of the selected page replacement algorithm in three parts: Current Frame, Added Page, and New Frame. The "Current Frame" shows the contents of memory before the new page is processed. "Added Page" highlights the current page being accessed, and "New Frame" reflects the memory frame contents after applying the page replacement logic. Color coding is used here to indicate whether a page was retained, added, or removed.

## Options

The options section provides the user with three buttons: FIFO, LRU, and Optimal. These represent the three page replacement algorithms available in the simulator. Selecting one of these algorithms determines how the simulation will manage page replacements during execution.

## Controls

This section provides the core buttons to interact with the simulation: Start, Next, and Clear. After setting up the configuration and selecting an algorithm, the user clicks Start to begin the simulation. The Next button steps through the reference string one page at a time, allowing the user to observe each change in memory. The Clear button resets all inputs and visual elements, allowing a fresh simulation to be run.

## Status & Info

This section displays useful runtime information. The Legend explains the meaning of the colors used in the simulation: red for removed pages, blue for added pages, and green for retained pages. Below the legend, the current Algorithm in use is displayed, along with indicators for Hit/Miss status and the total number of Page Faults encountered during the simulation. A completion label also appears at the end of the simulation to signal that the reference string has been fully processed. This is shown on Figure 2.

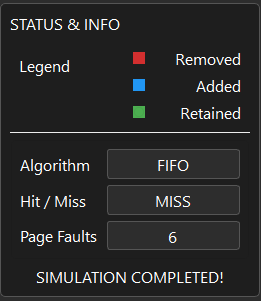


Figure 2: The complete Status & Info section after the simulation is completed

# Sample Outputs

To demonstrate the behavior and characteristics of each page replacement algorithm, here are sample simulations with identical input parameters.

## Simulation 1 [ 6, 3, 6, 9, 5, 5, 0, 5, 4, 2, 7, 0 ]

### FIFO Algorithm Outputs

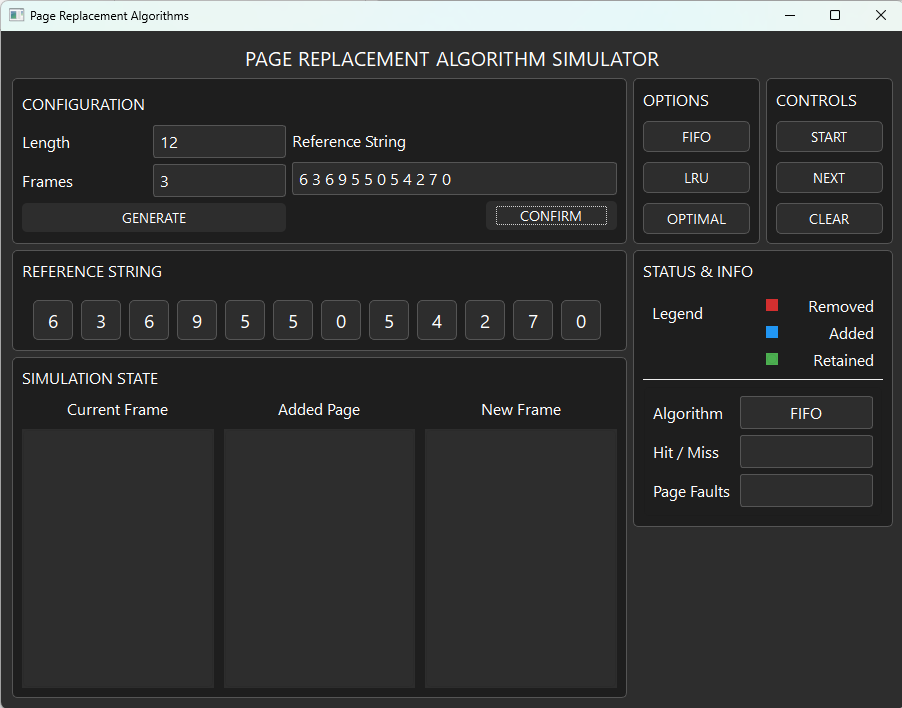


Figure 3.1: Initial Output for FIFO Simulation 1

This shows the initial output before the start button has been clicked. As it can be seen in the image the provided length and frames are 12 and 3 respectively. The reference string generated is also shown on the reference string section and the FIFO algorithm is also shown on the status & info section.

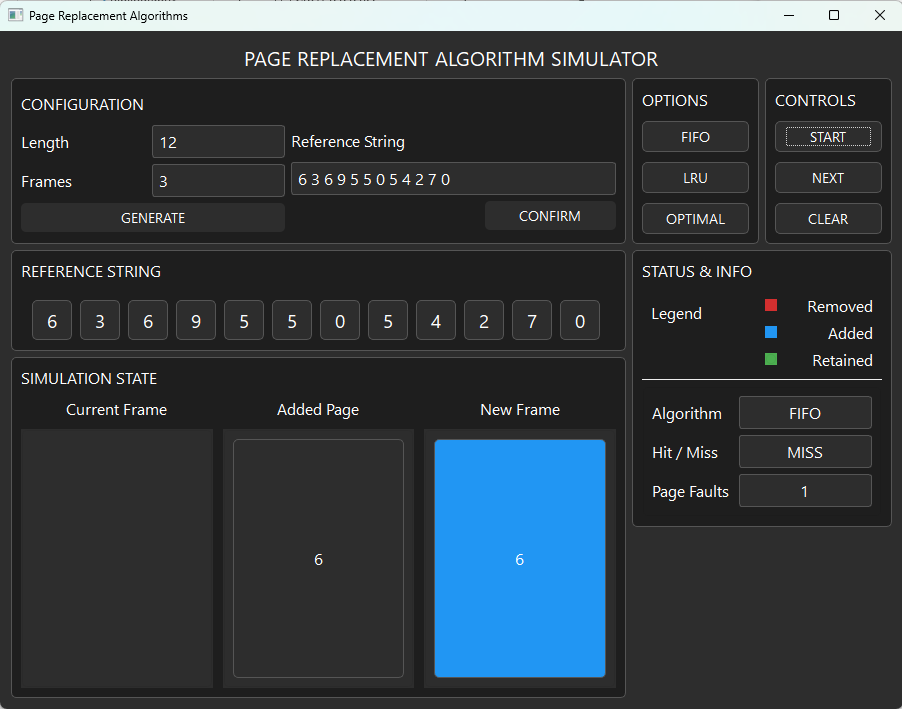


Figure 3.2: Output of FIFO Simulation 1 after the Start Button is Pressed

This shows the output after the start button is clicked. It shows that the current frame has no page yet because the simulation has just started. Meanwhile, on the added page it shows the page [6] which is the first page of our reference string and on the new frame it showed [6] in a blue frame because as the legend indicates it is an added frame. We can also see that it says that it was a miss and with that the page fault counter increased to 1.

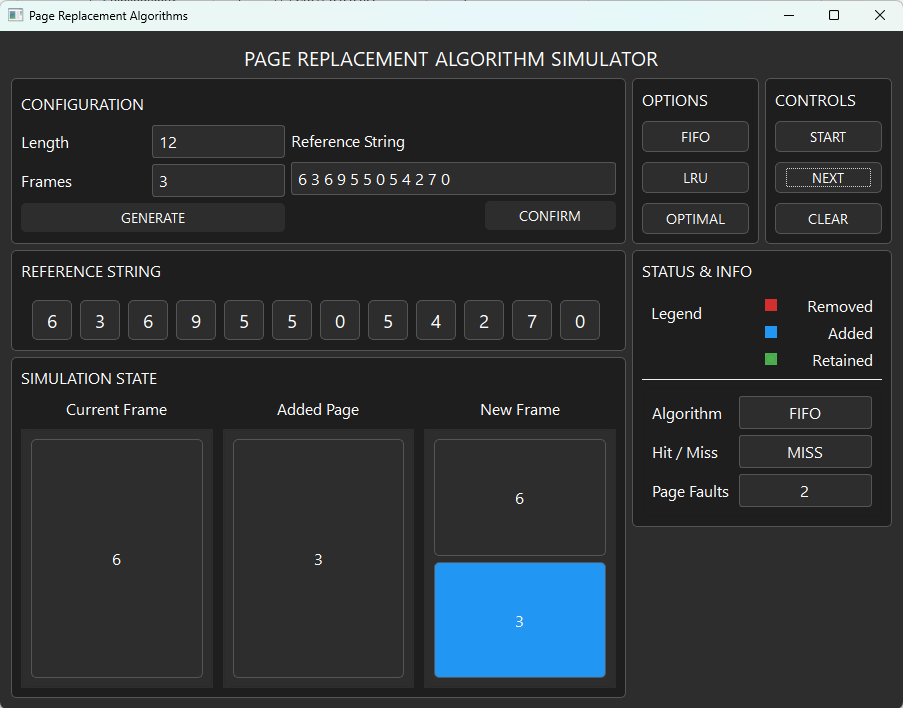


Figure 3.3: The second step on the FIFO Simulation 1

In here, the second page reference which is [3] is added. The simulation state section shows the current frame has [6] and the added page is [3] which results to the new frame having [6][3] with [3] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 2.

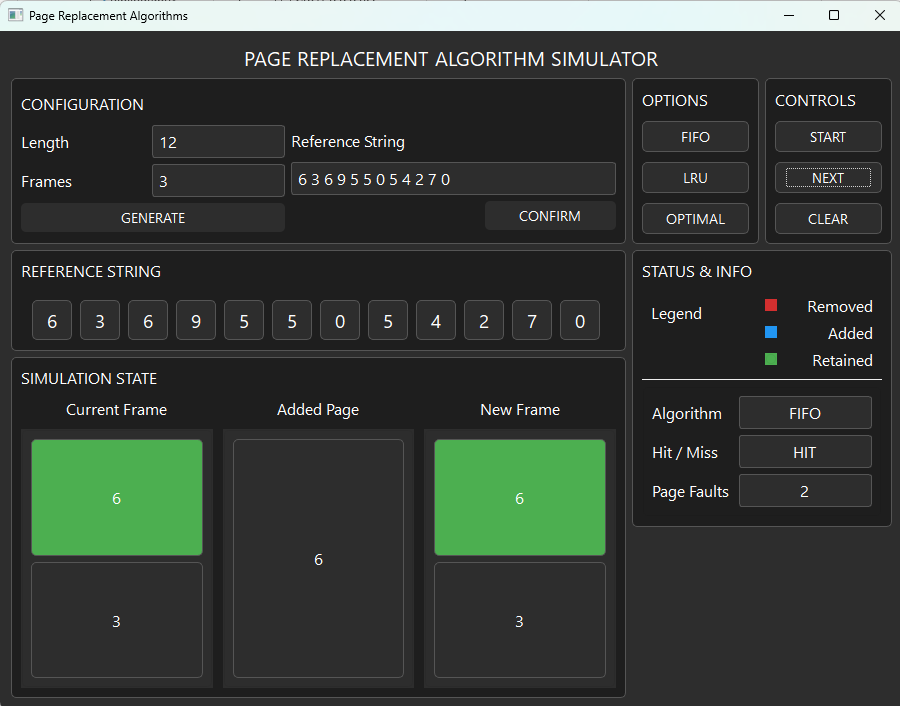


Figure 3.4: The third step on the FIFO simulation 1

In here, the third page reference which is [6] is being added which is shown on the added page section. It also shows the current frame and the new frame having [6][3] with [6] in a green frame because the added page is also a [6] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 2.

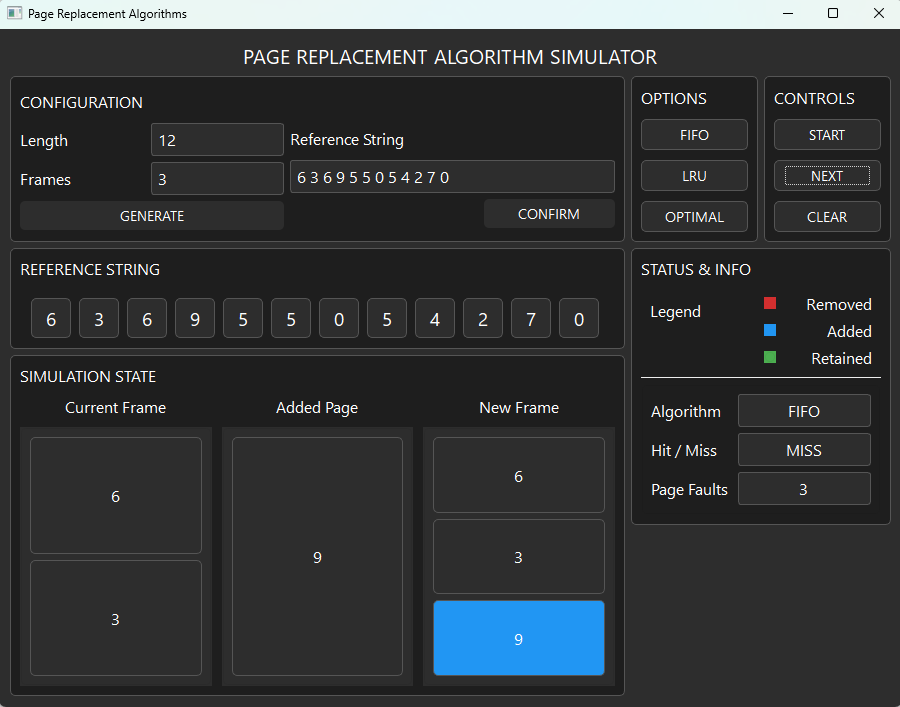


Figure 3.5: The fourth step on the FIFO simulation 1

In here, the fourth page reference which is [9] is added. The simulation state section shows the current frame has [6][3] and the added page is [9] which results to the new frame having [6][3][9] with [9] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 3.

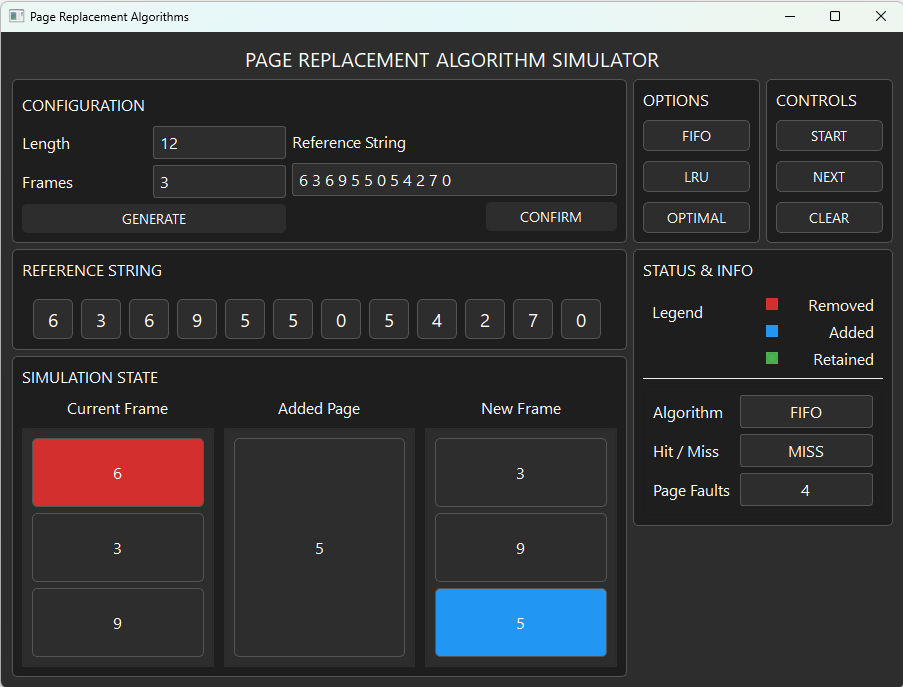


Figure 3.6: The fifth step on the FIFO simulation 1

In here, the fifth page reference which is [5] is added. The simulation state section shows the current frame has [6][3][9] with [6] in a red frame because it is being removed and the added page is [5] which results to the new frame having [3][9][5] with [5] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 4.

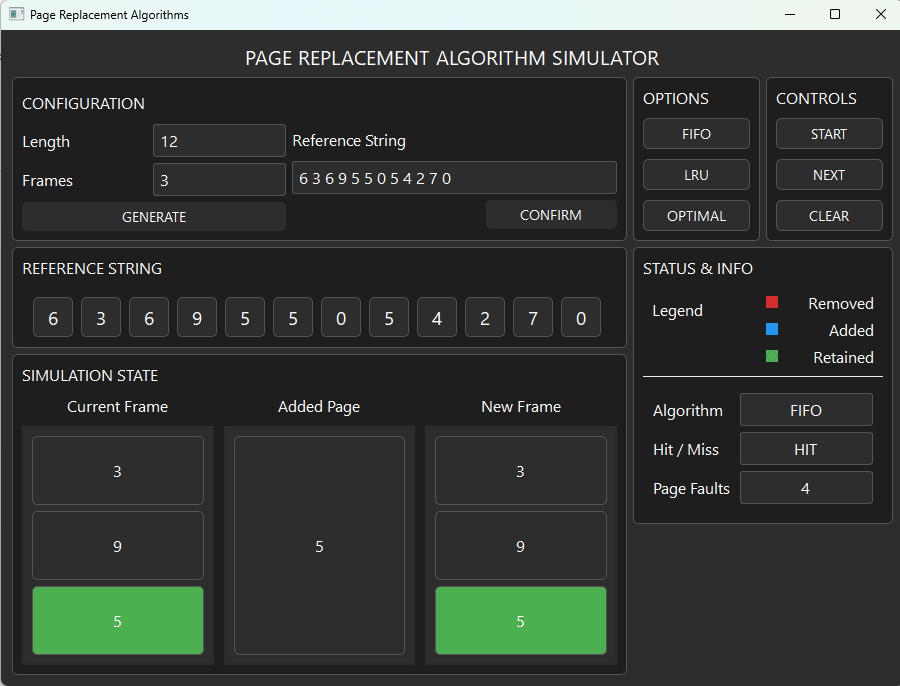


Figure 3.7: The sixth step on the FIFO simulation 1

In here, the sixth page reference which is [5] is being added which is shown on the added page section. It also shows the current frame and the new frame having [3][9][5] with [5] in a green frame because the added page is also a [5] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 4.

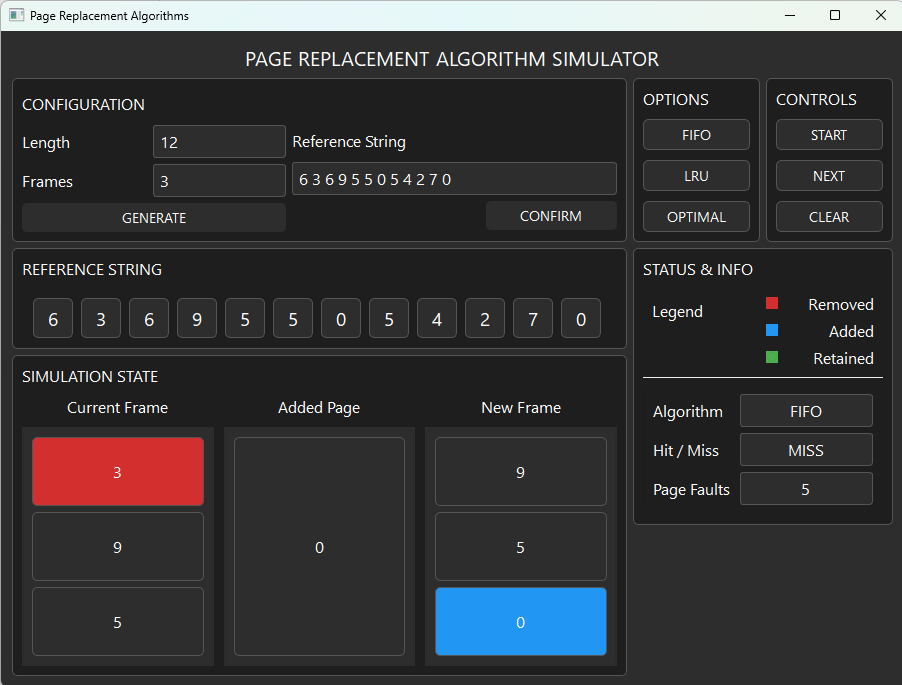


Figure 3.8: The seventh step on the FIFO simulation 1

In here, the seventh page reference which is [0] is added. The simulation state section shows the current frame has [3][9][5] with [3] in a red frame because it is being removed and the added page is [0] which results to the new frame having [9][5][0] with [0] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 5.

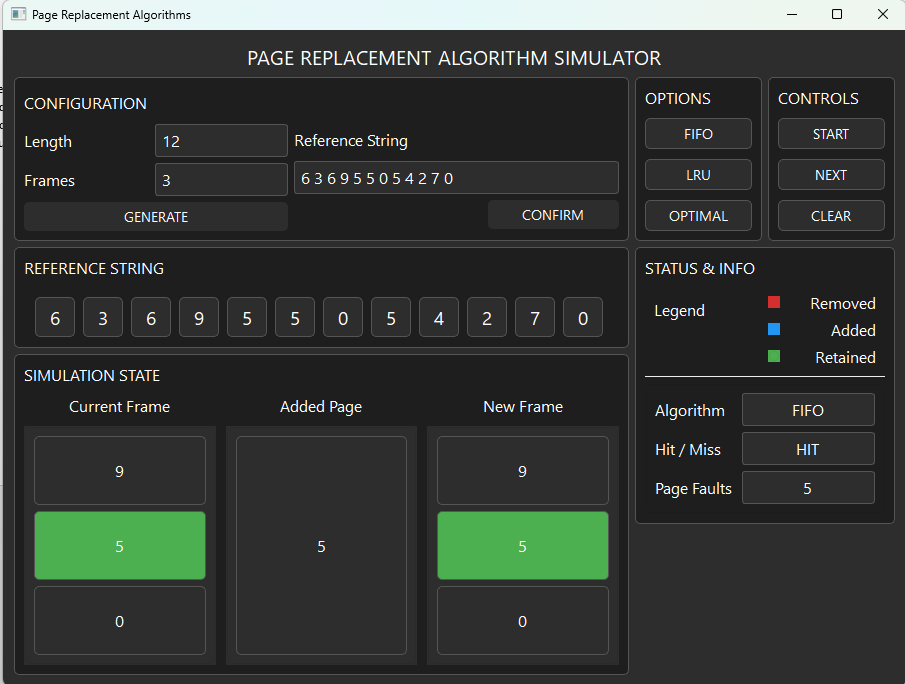


Figure 3.9: The eighth step on the FIFO simulation 1

In here, the eighth page reference which is [5] is being added which is shown on the added page section. It also shows the current frame and the new frame having [9][5][0] with [5] in a green frame because the added page is also a [5] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 5.

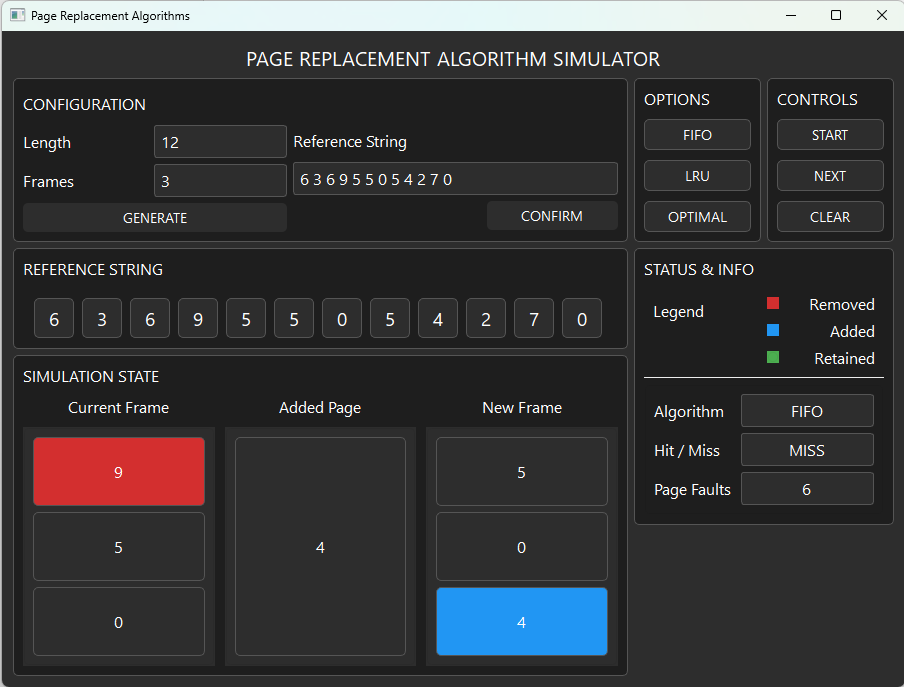


Figure 3.10: The ninth step on the FIFO simulation 1

In here, the ninth page reference which is [4] is added. The simulation state section shows the current frame has [9][5][0] with [9] in a red frame because it is being removed and the added page is [4] which results to the new frame having [5][0][4] with [4] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 6.

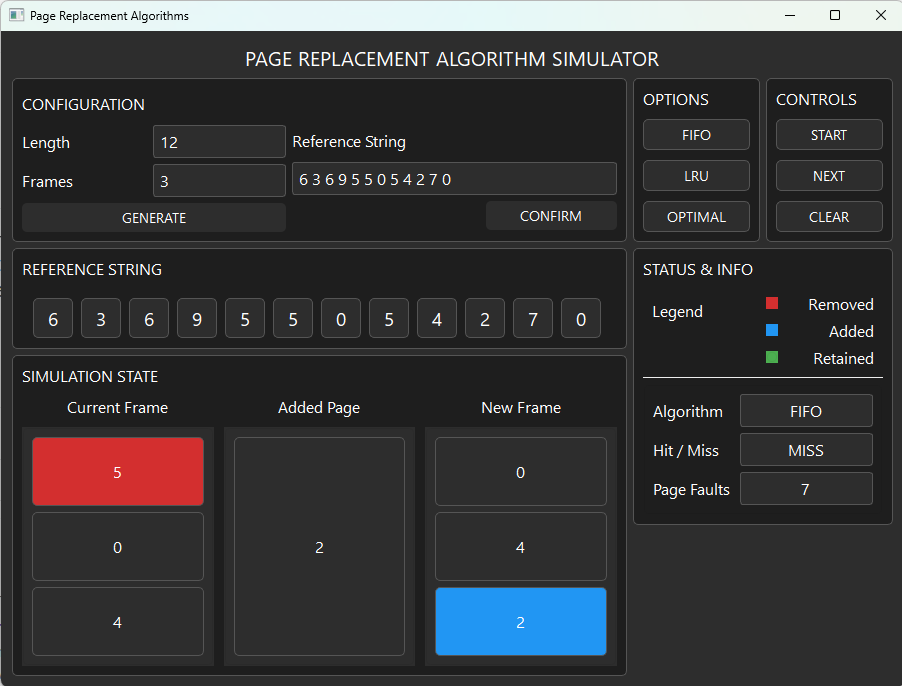


Figure 3.11: The tenth step on the FIFO simulation 1

In here, the tenth page reference which is [2] is added. The simulation state section shows the current frame has [5][0][4] with [5] in a red frame because it is being removed and the added page is [2] which results to the new frame having [0][4][2] with [2] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 7.

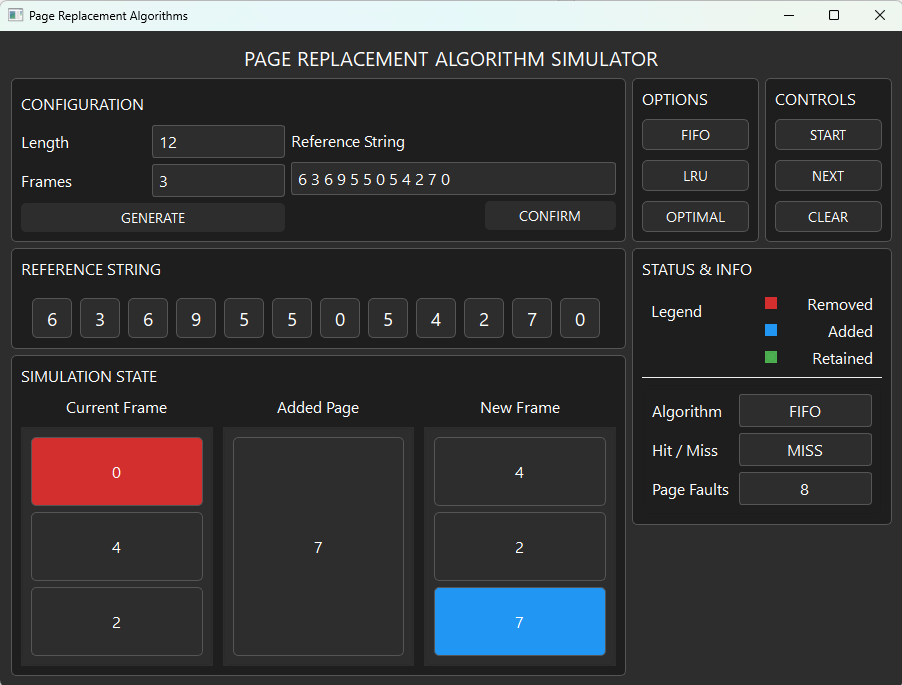


Figure 3.12: The eleventh step on the FIFO simulation 1

In here, the eleventh page reference which is [7] is added. The simulation state section shows the current frame has [0][4][2] with [0] in a red frame because it is being removed and the added page is [7] which results to the new frame having [4][2][7] with [7] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 8.

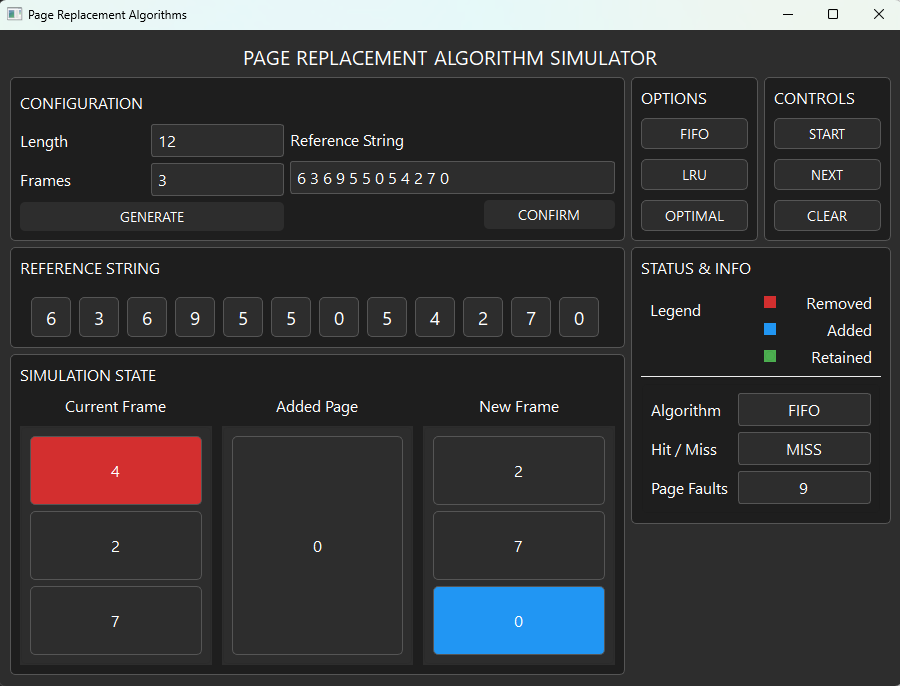


Figure 3.13: The twelfth step on the FIFO simulation 1

In here, the twelfth page reference which is [0] is added. The simulation state section shows the current frame has [4][2][7] with [4] in a red frame because it is being removed and the added page is [0] which results to the new frame having [2][7][0] with [0] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 9.

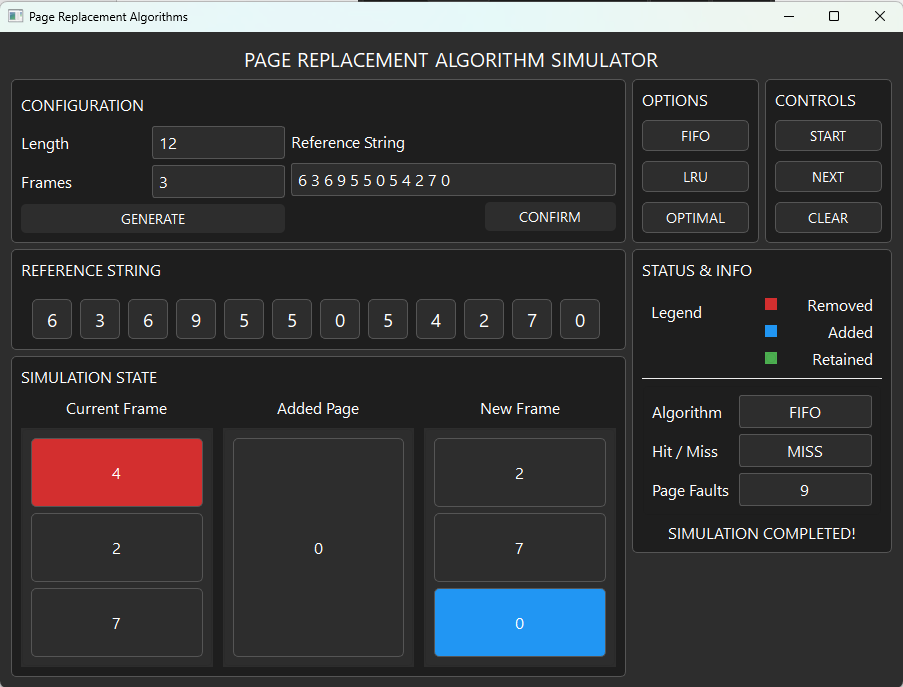


Figure 3.14: The thirteenth step on the FIFO simulation 1

This shows that the Simulation Completed text has been visible which states that our simulation has finished and the final number of page faults in this simulation is 9.

### LRU Algorithm Outputs

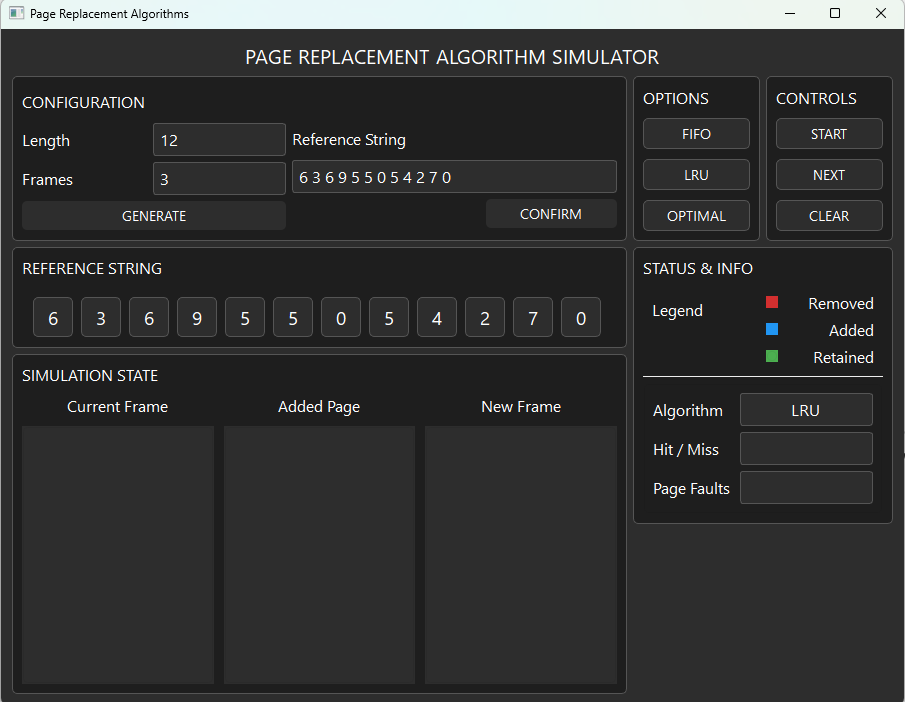


Figure 4.1: Initial Output for LRU Simulation 1

This shows the initial output before the start button has been clicked. As it can be seen in the image the provided length and frames are 12 and 3 respectively. The reference string generated is also shown on the reference string section and the LRU algorithm is also shown on the status & info section.

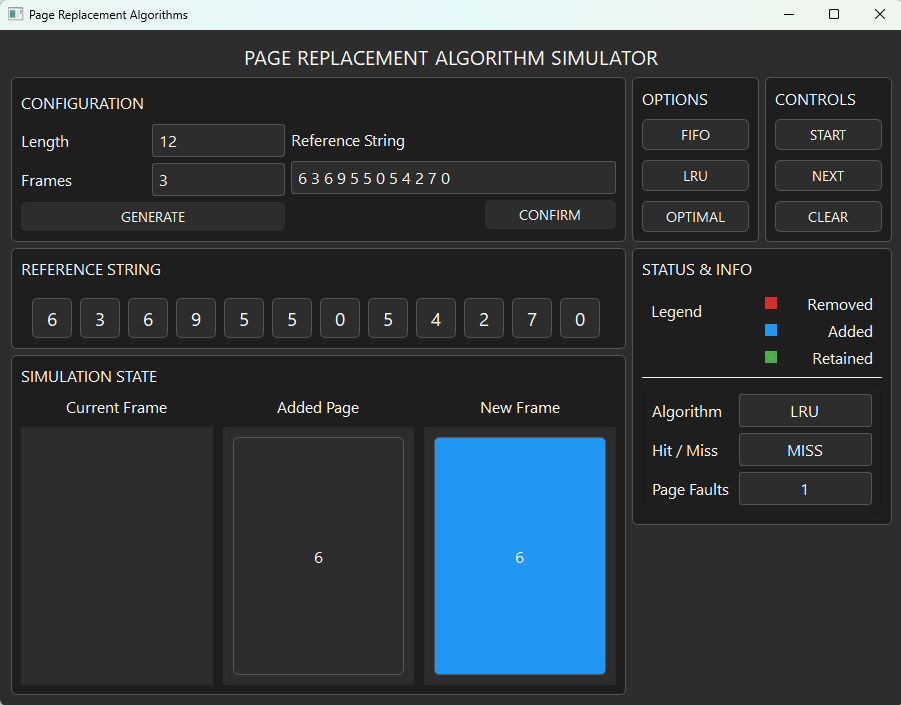


Figure 4.2: Output of LRU Simulation 1 after the Start Button is Pressed

This shows the output after the start button is clicked. It shows that the current frame has no page yet because the simulation has just started. Meanwhile, on the added page it shows the page [6] which is the first page of our reference string and on the new frame it showed [6] in a blue frame because as the legend indicates it is an added frame. We can also see that it says that it was a miss and with that the page fault counter increased to 1.

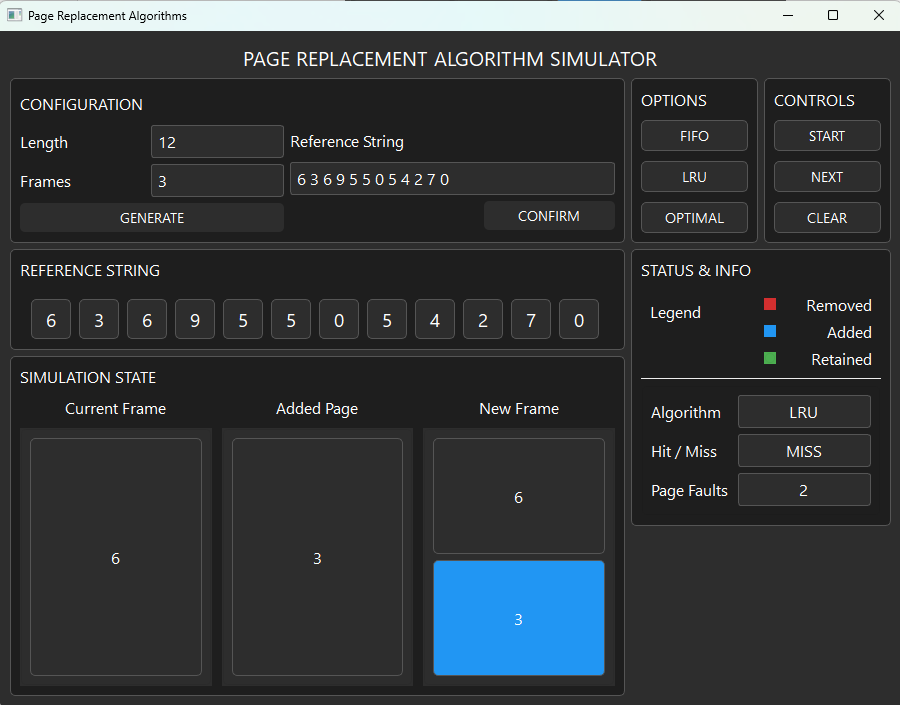


Figure 4.3: The second step on the LRU Simulation 1

In here, the second page reference which is [3] is added. The simulation state section shows the current frame has [6] and the added page is [3] which results to the new frame having [6][3] with [3] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 2.

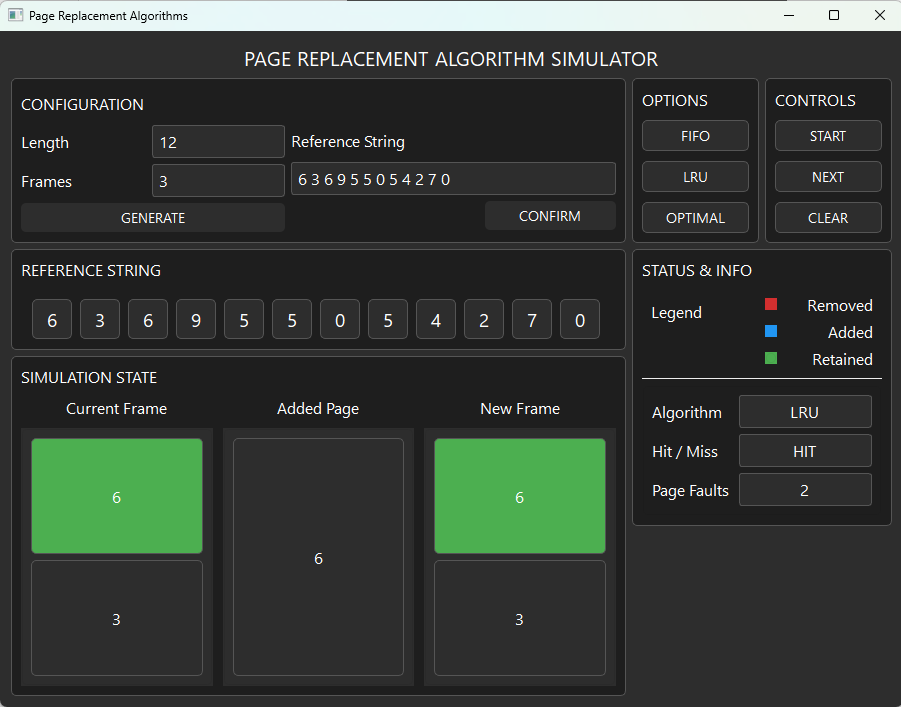


Figure 4.4: The third step on the LRU simulation 1

In here, the third page reference which is [6] is being added which is shown on the added page section. It also shows the current frame and the new frame having [6][3] with [6] in a green frame because the added page is also a [6] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 2.

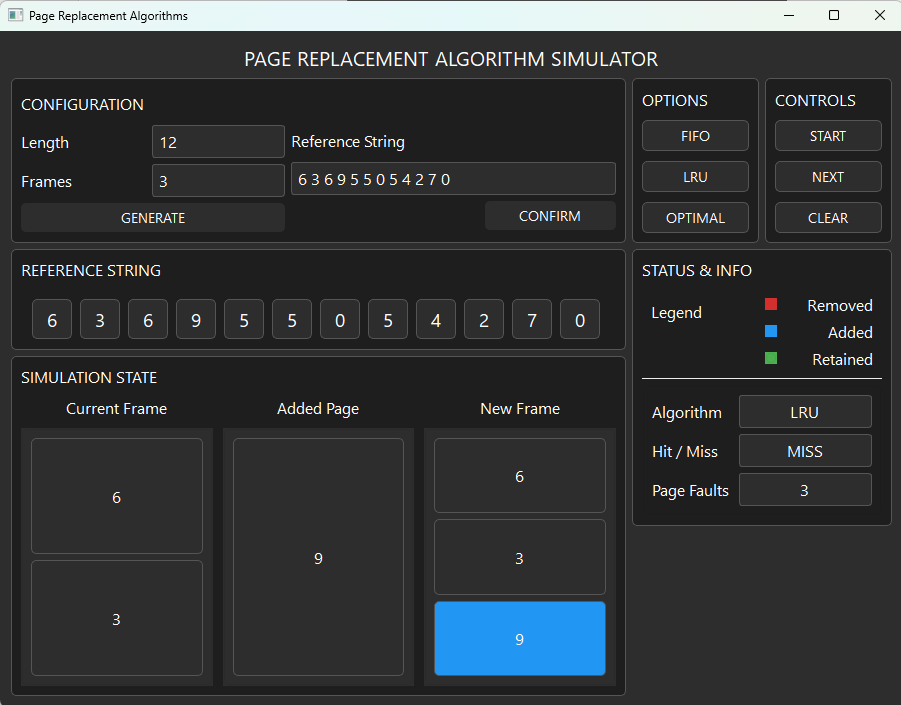


Figure 4.5: The fourth step on the LRU simulation 1

In here, the fourth page reference which is [9] is added. The simulation state section shows the current frame has [6][3] and the added page is [9] which results to the new frame having [6][3][9] with [9] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 3.

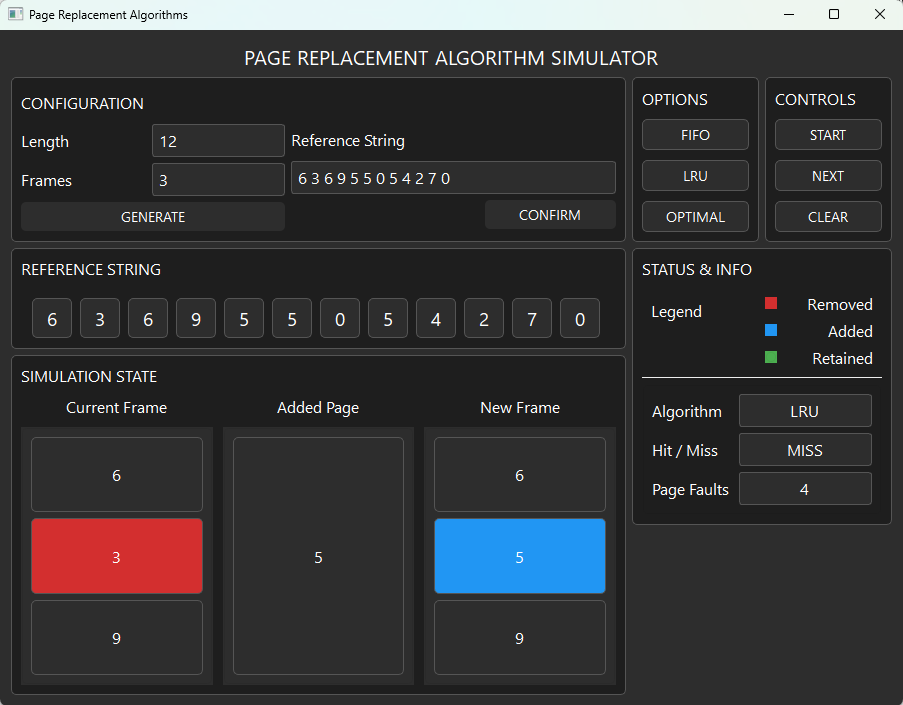


Figure 4.6: The fifth step on the LRU simulation 1

In here, the fifth page reference which is [5] is added. The simulation state section shows the current frame has [6][3][9] with [3] in a red frame because it is being removed and the added page is [5] which results to the new frame having [6][5][9] with [5] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 4.

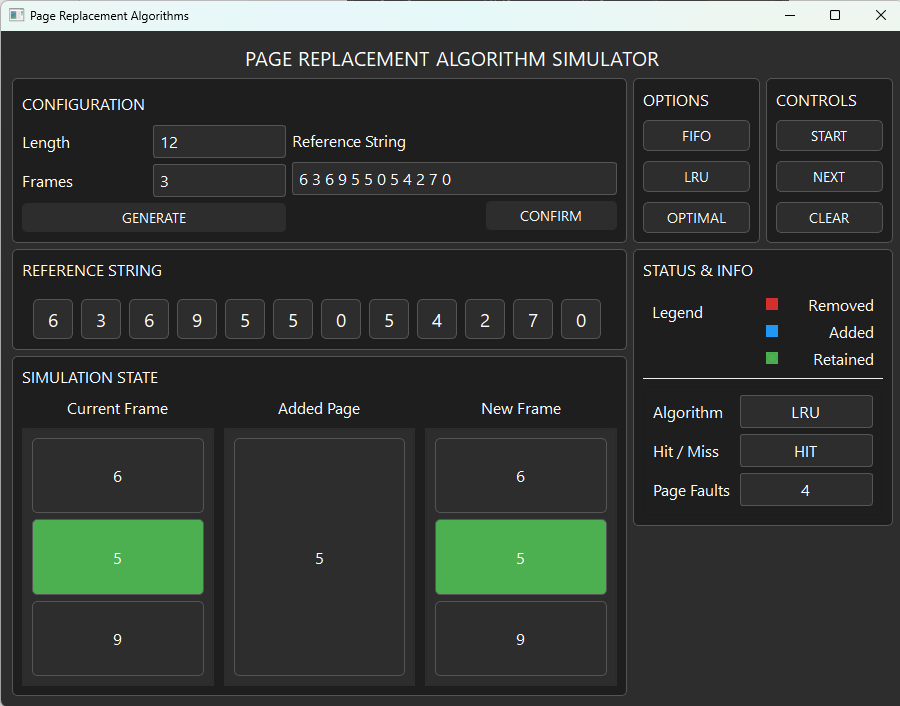


Figure 4.7: The sixth step on the LRU simulation 1

In here, the sixth page reference which is [5] is being added which is shown on the added page section. It also shows the current frame and the new frame having [6][5][9] with [5] in a green frame because the added page is also a [5] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 4.

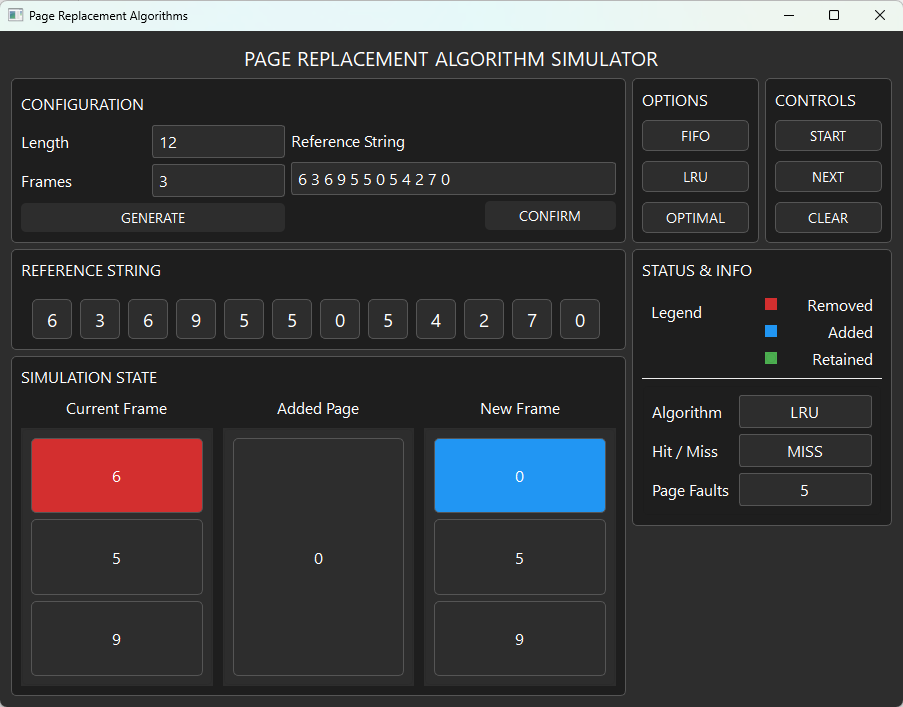


Figure 4.8: The seventh step on the LRU simulation 1

In here, the seventh page reference which is [0] is added. The simulation state section shows the current frame has [6][5][9] with [6] in a red frame because it is being removed and the added page is [0] which results to the new frame having [0][5][9] with [0] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 5.

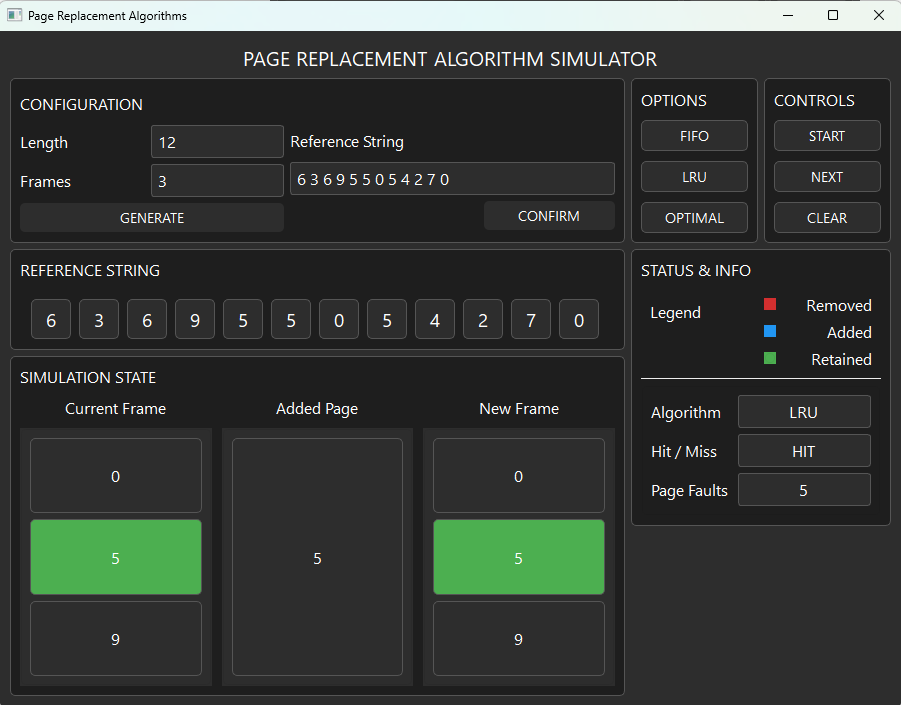


Figure 4.9: The eighth step on the LRU simulation 1

In here, the eighth page reference which is [5] is being added which is shown on the added page section. It also shows the current frame and the new frame having [0][5][9] with [5] in a green frame because the added page is also a [5] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 5.

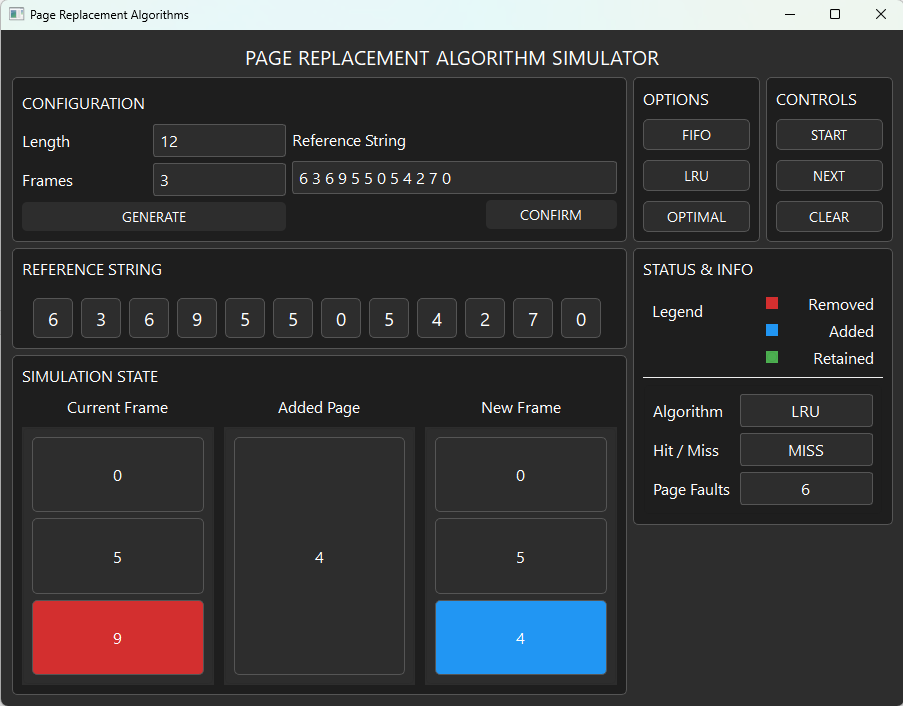


Figure 4.10: The ninth step on the LRU simulation 1

In here, the ninth page reference which is [4] is added. The simulation state section shows the current frame has [0][5][9] with [9] in a red frame because it is being removed and the added page is [4] which results to the new frame having [0][5][4] with [4] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 6.

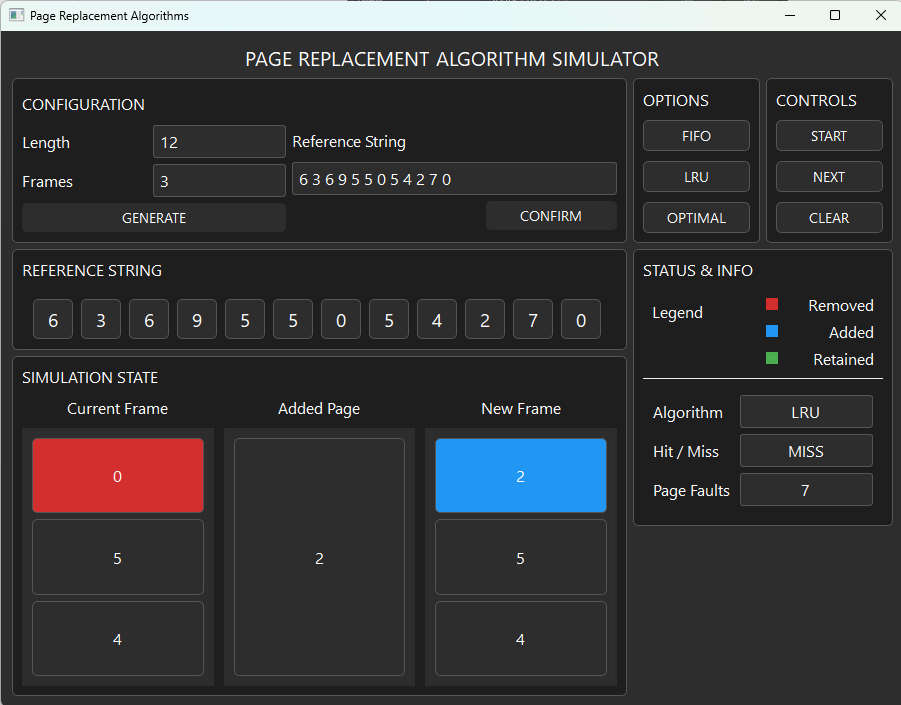


Figure 4.11: The tenth step on the LRU simulation 1

In here, the tenth page reference which is [2] is added. The simulation state section shows the current frame has [0][5][4] with [0] in a red frame because it is being removed and the added page is [2] which results to the new frame having [2][5][4] with [2] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 7.

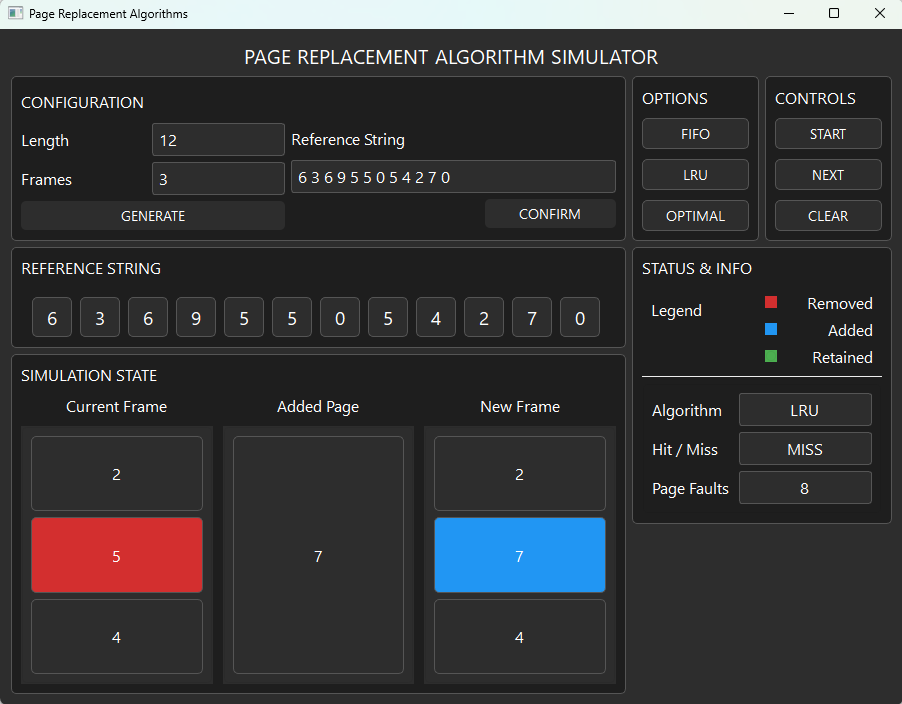


Figure 4.12: The eleventh step on the LRU simulation 1

In here, the eleventh page reference which is [7] is added. The simulation state section shows the current frame has [2][5][4] with [5] in a red frame because it is being removed and the added page is [7] which results to the new frame having [2][7][4] with [7] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 8.

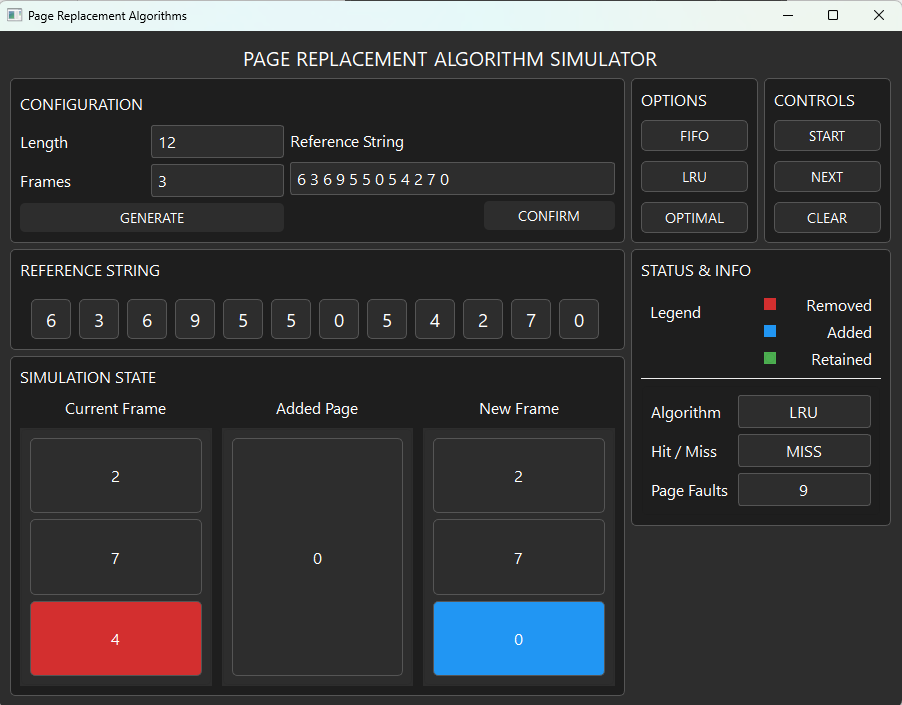


Figure 4.13: The twelfth step on the LRU simulation 1

In here, the twelfth page reference which is [0] is added. The simulation state section shows the current frame has [2][7][4] with [4] in a red frame because it is being removed and the added page is [0] which results to the new frame having [2][7][0] with [0] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 9.

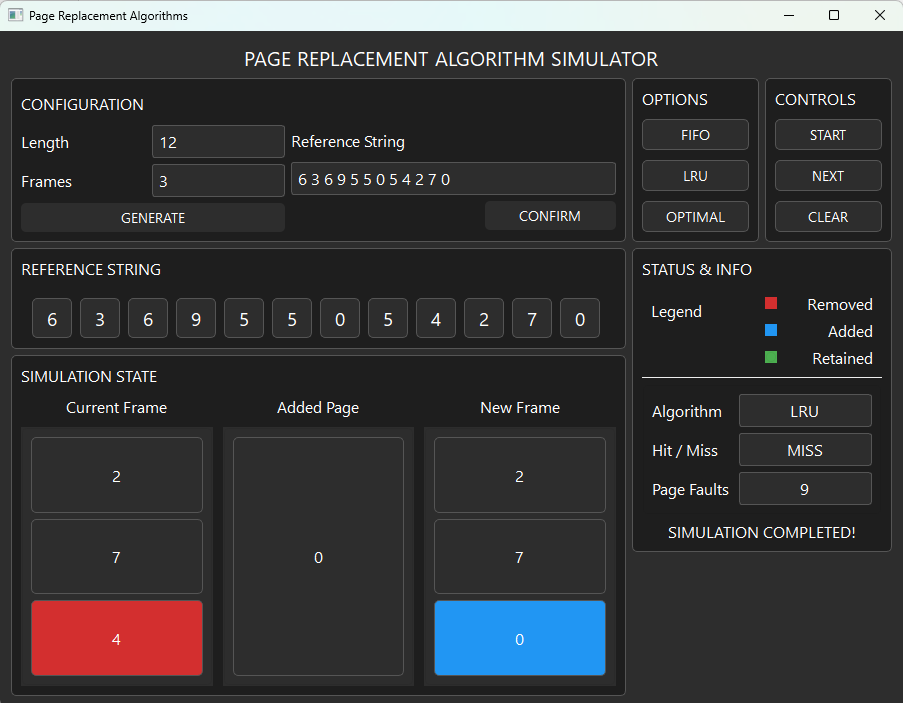


Figure 4.14: The thirteenth step on the LRU simulation 1

This shows that the Simulation Completed text has been visible which states that our simulation has finished and the final number of page faults in this simulation is 9.

### Optimal Algorithm Outputs

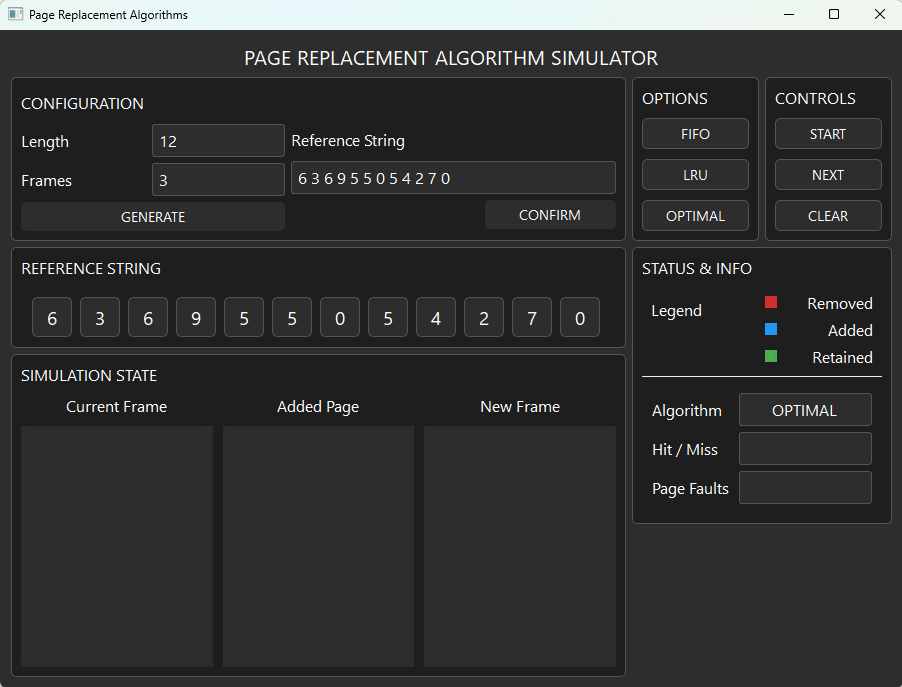


Figure 5.1: Initial Output for Optimal Simulation 1

This shows the initial output before the start button has been clicked. As it can be seen in the image the provided length and frames are 12 and 3 respectively. The reference string generated is also shown on the reference string section and the Optimal algorithm is also shown on the status & info section.

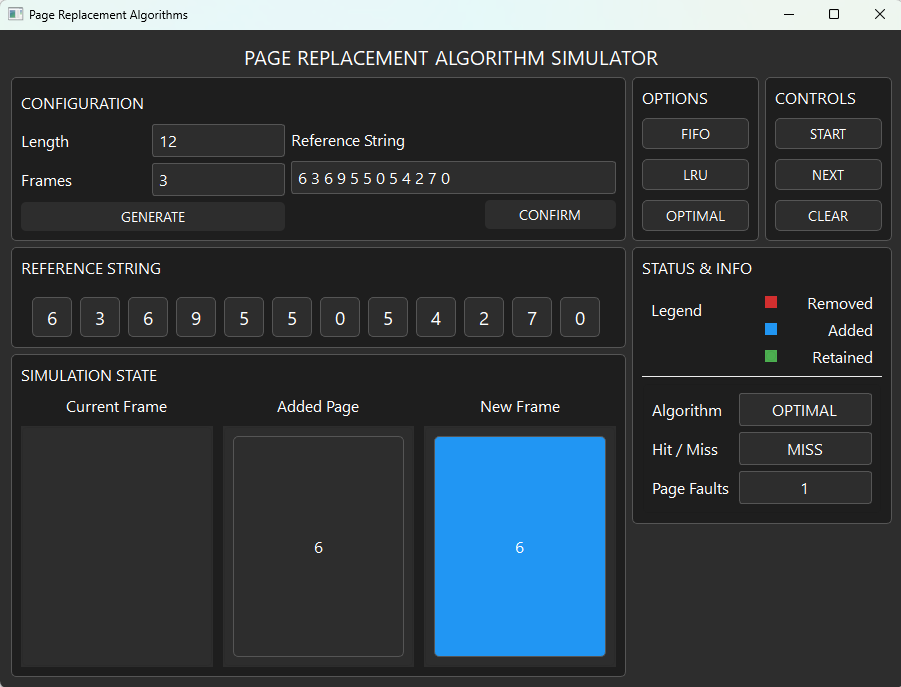


Figure 5.2: Output of Optimal Simulation 1 after the Start Button is Pressed

This shows the output after the start button is clicked. It shows that the current frame has no page yet because the simulation has just started. Meanwhile, on the added page it shows the page [6] which is the first page of our reference string and on the new frame it showed [6] in a blue frame because as the legend indicates it is an added frame. We can also see that it says that it was a miss and with that the page fault counter increased to 1.

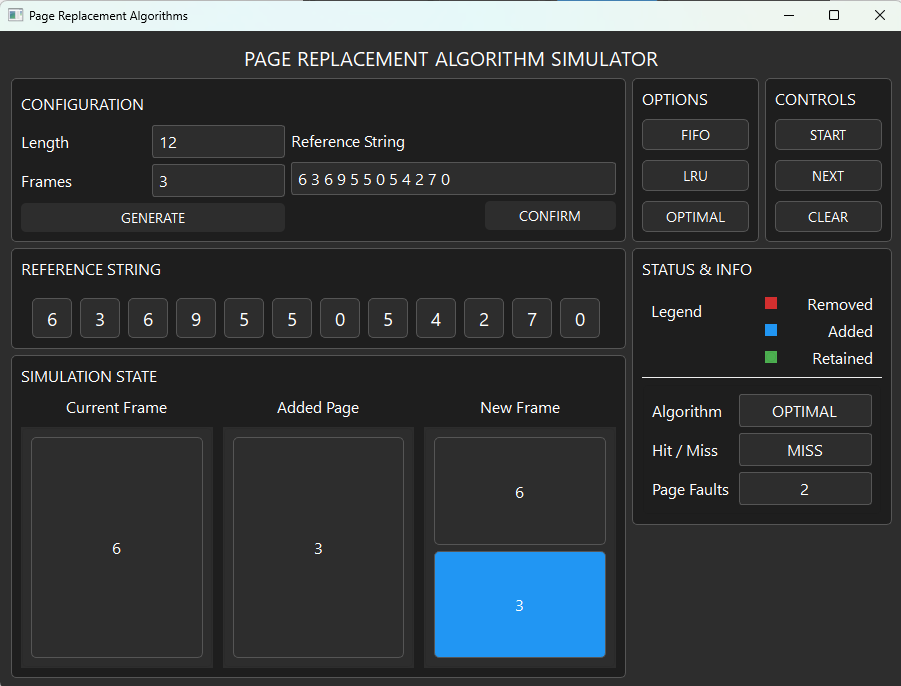


Figure 5.3: The second step on the Optimal Simulation 1

In here, the second page reference which is [3] is added. The simulation state section shows the current frame has [6] and the added page is [3] which results to the new frame having [6][3] with [3] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 2.

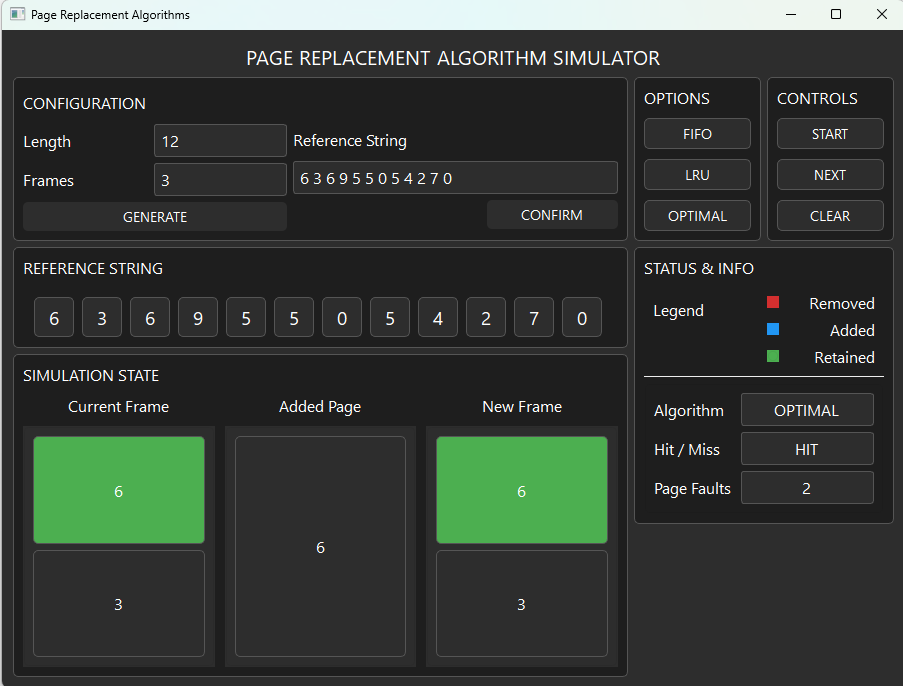


Figure 5.4: The third step on the Optimal Simulation 1

In here, the third page reference which is [6] is being added which is shown on the added page section. It also shows the current frame and the new frame having [6][3] with [6] in a green frame because the added page is also a [6] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 2.

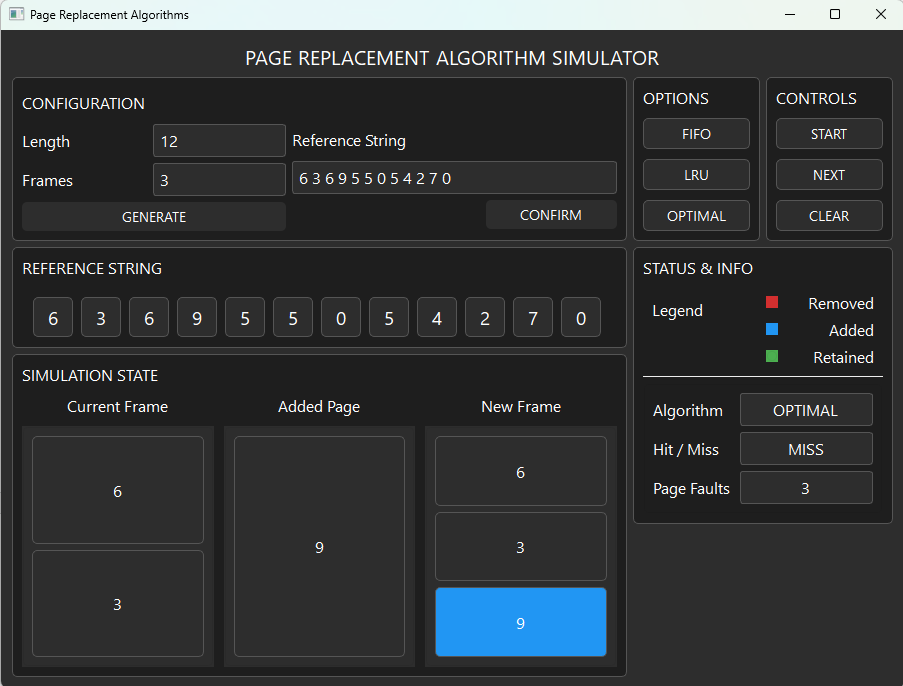


Figure 5.5: The fourth step on the Optimal Simulation 1

In here, the fourth page reference which is [9] is added. The simulation state section shows the current frame has [6][3] and the added page is [9] which results to the new frame having [6][3][9] with [9] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 3.

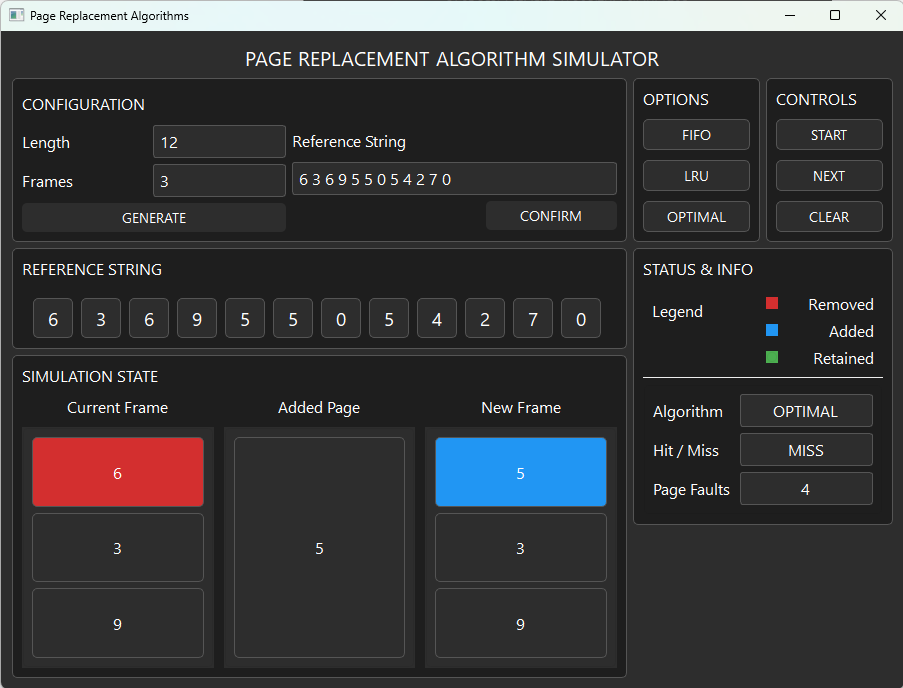


Figure 5.6: The fifth step on the Optimal Simulation 1

In here, the fifth page reference which is [5] is added. The simulation state section shows the current frame has [6][3][9] with [6] in a red frame because it is being removed and the added page is [5] which results to the new frame having [3][9][5] with [5] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 4.

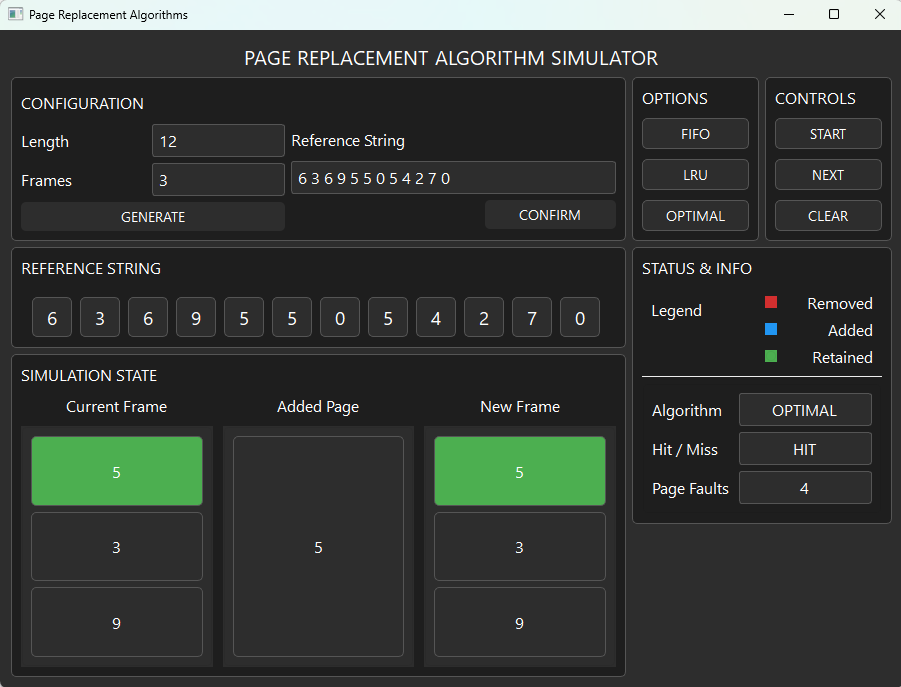


Figure 5.7: The sixth step on the Optimal Simulation 1

In here, the sixth page reference which is [5] is being added which is shown on the added page section. It also shows the current frame and the new frame having [5][3][9] with [5] in a green frame because the added page is also a [5] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 4.

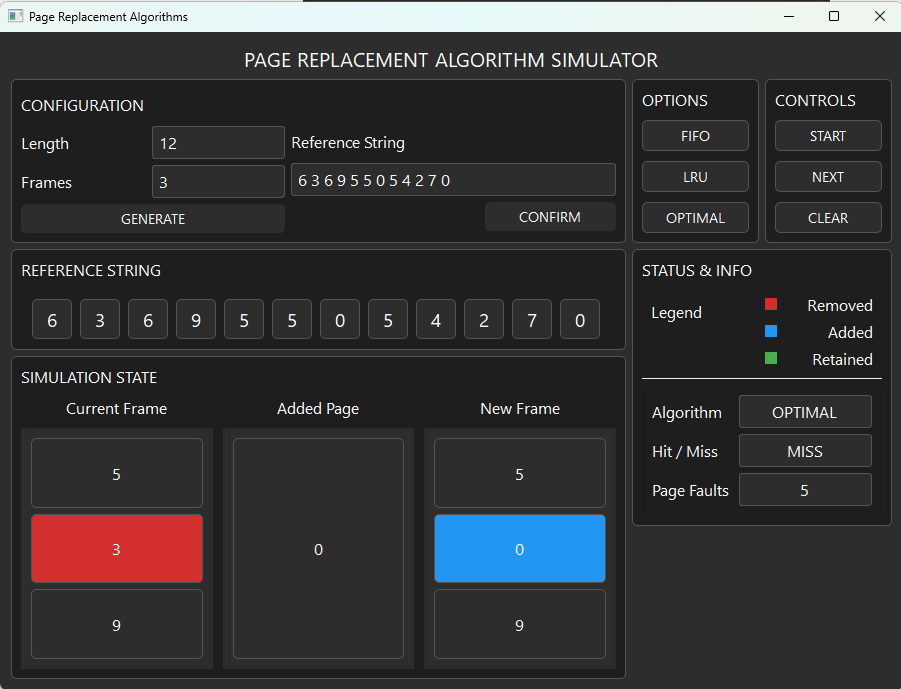


Figure 5.8: The seventh step on the Optimal Simulation 1

In here, the seventh page reference which is [0] is added. The simulation state section shows the current frame has [5][3][9] with [3] in a red frame because it is being removed and the added page is [0] which results to the new frame having [5][0][9] with [0] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 5.

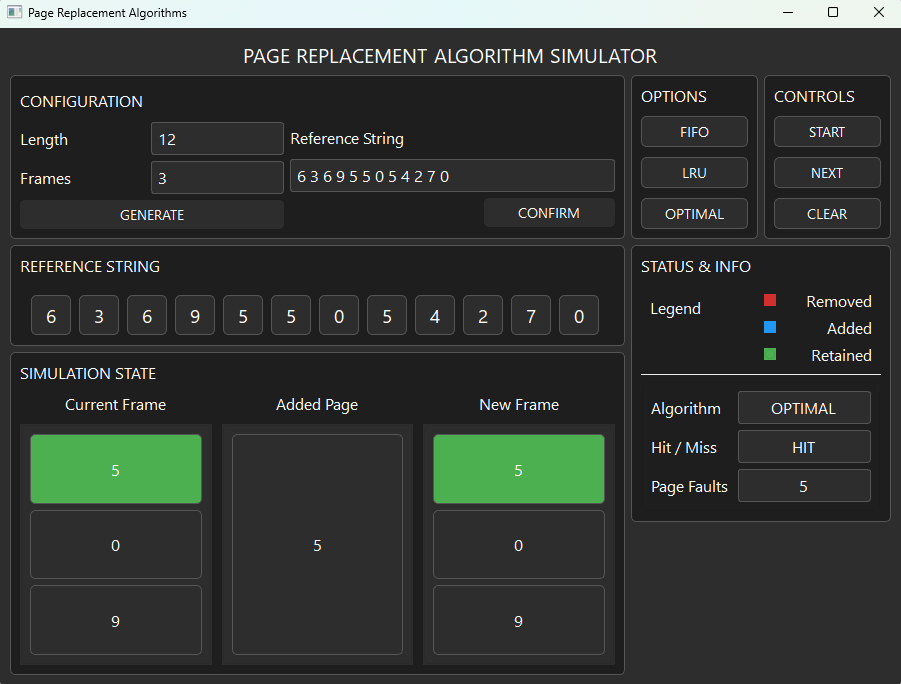


Figure 5.9: The eighth step on the Optimal Simulation 1

In here, the sixth page reference which is [5] is being added which is shown on the added page section. It also shows the current frame and the new frame having [5][0][9] with [5] in a green frame because the added page is also a [5] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 5.

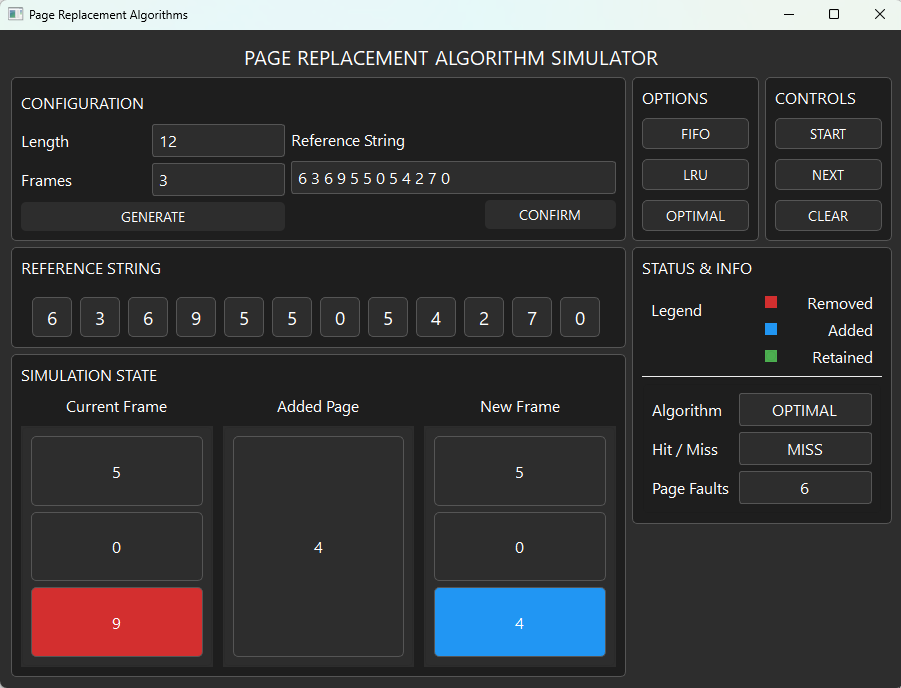


Figure 5.10: The ninth step on the Optimal Simulation 1

In here, the ninth page reference which is [4] is added. The simulation state section shows the current frame has [5][0][9] with [9] in a red frame because it is being removed and the added page is [4] which results to the new frame having [5][0][4] with [4] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 6.

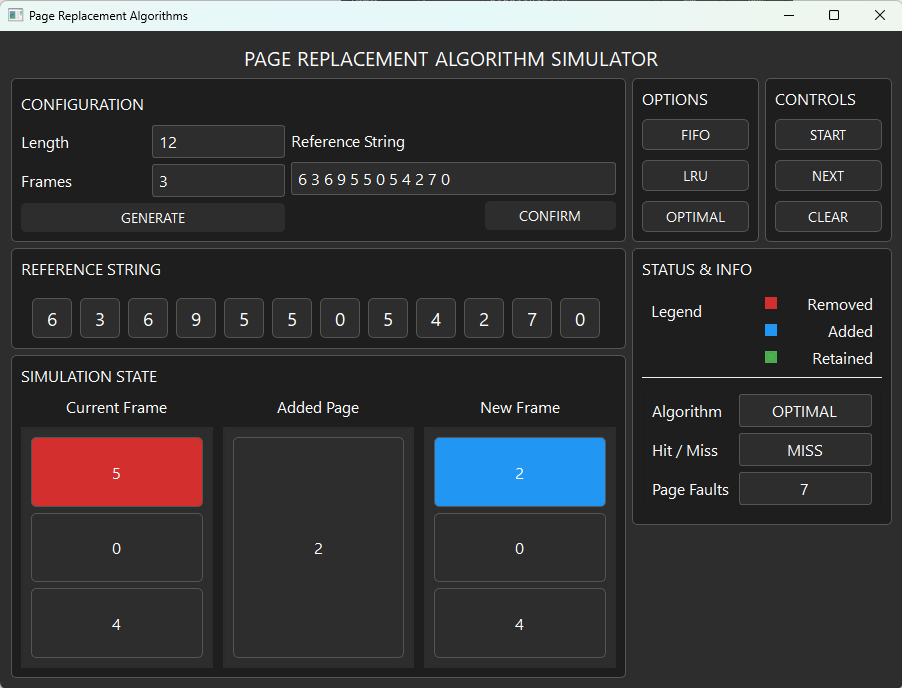


Figure 5.11: The tenth step on the Optimal Simulation 1

In here, the tenth page reference which is [2] is added. The simulation state section shows the current frame has [5][0][4] with [5] in a red frame because it is being removed and the added page is [2] which results to the new frame having [2][0][4] with [2] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 7.

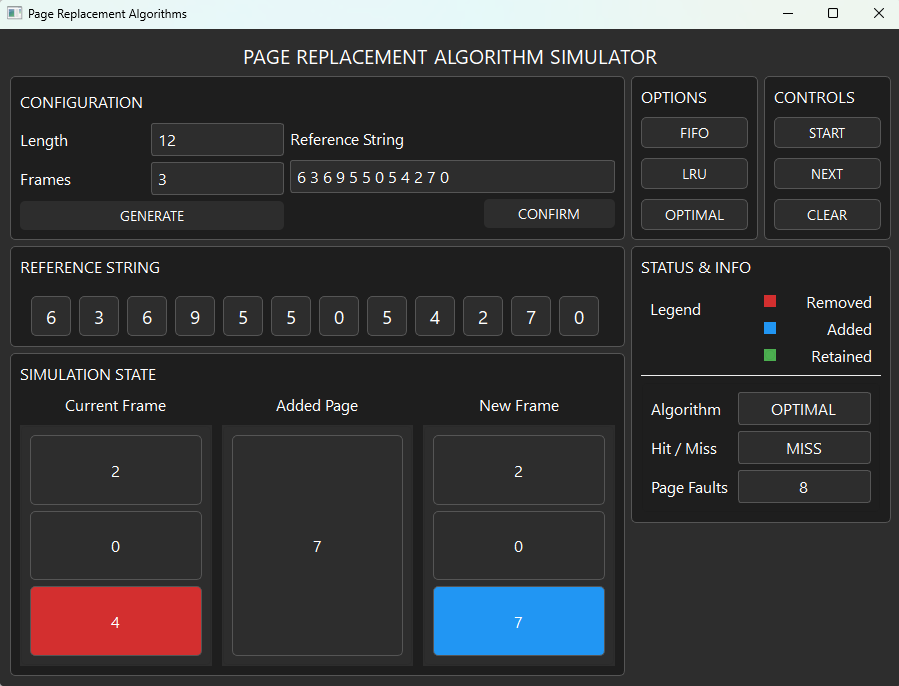


Figure 5.12: The eleventh step on the Optimal Simulation 1

In here, the eleventh page reference which is [7] is added. The simulation state section shows the current frame has [2][0][4] with [4] in a red frame because it is being removed and the added page is [7] which results to the new frame having [2][0][7] with [7] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 8.

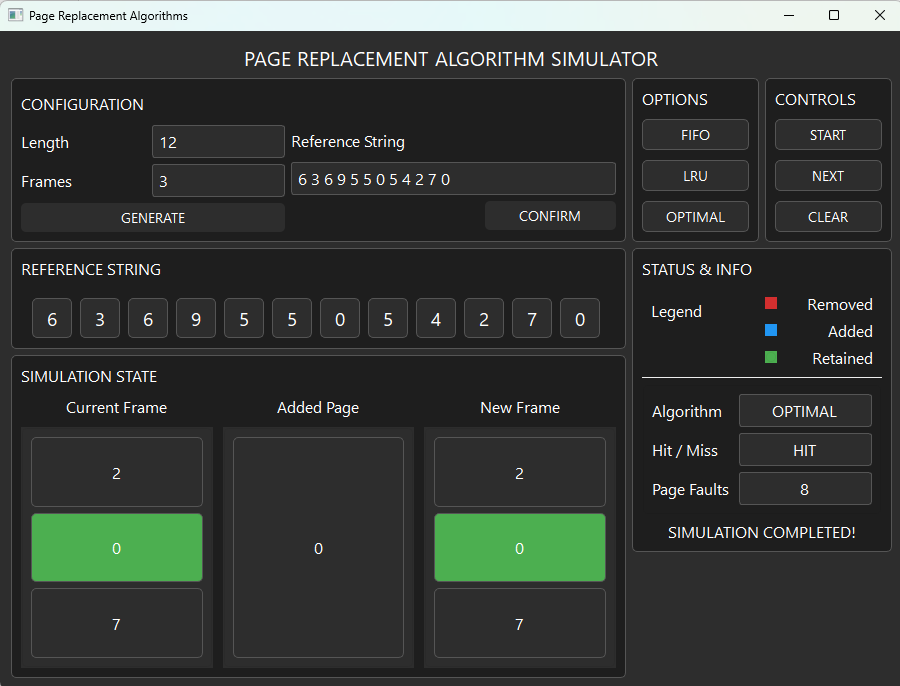


Figure 5.13: The twelfth step on the Optimal Simulation 1

In here, the twelfth page reference which is [0] is being added which is shown on the added page section. It also shows the current frame and the new frame having [2][0][7] with [0] in a green frame because the added page is also a [0] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 8. It also shows that the Simulation Completed text has been visible which states that our simulation has finished and the final number of page faults in this simulation is 8.

## Simulation 2 [ 5, 3, 2, 5, 9, 3, 8 ]

### FIFO Algorithm Outputs

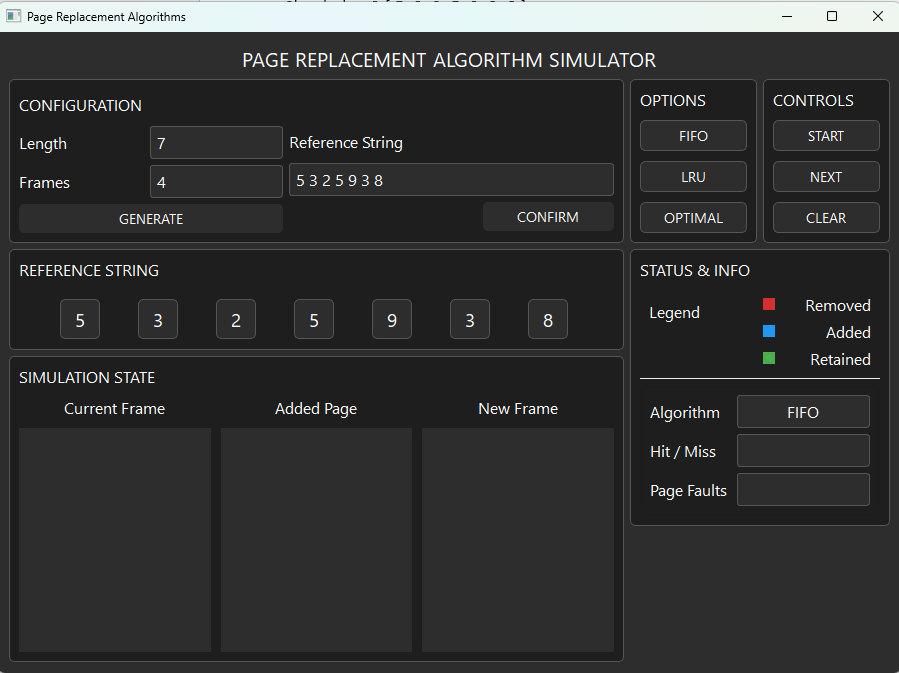


Figure 6.: Initial Output for FIFO Simulation 2

This shows the initial output before the start button has been clicked. As it can be seen in the image the provided length and frames are 7 and 4 respectively. The reference string generated is also shown on the reference string section and the FIFO algorithm is also shown on the status & info section.

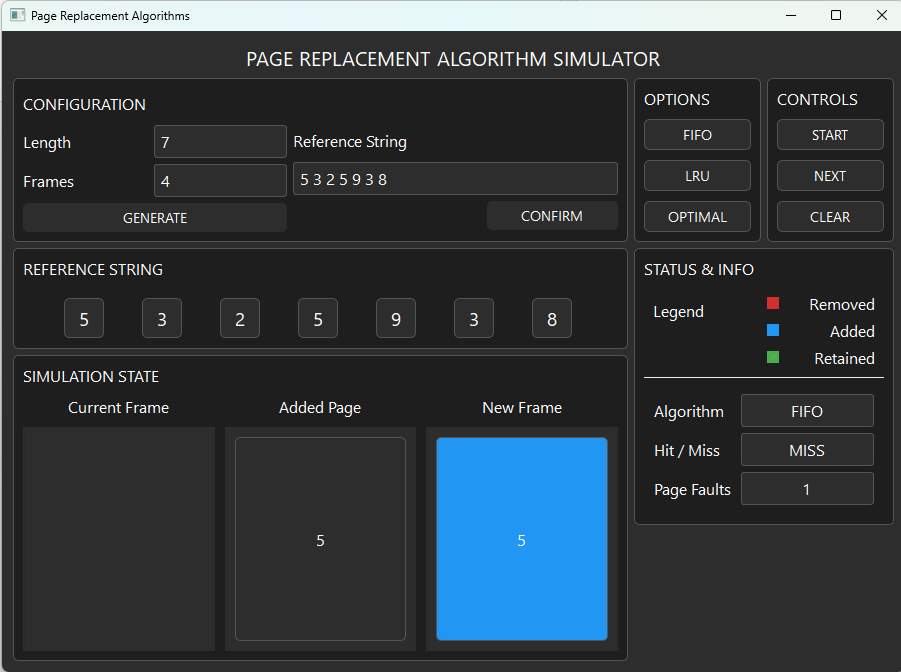


Figure 6.: Output of FIFO Simulation 2 after the Start Button is Pressed

This shows the output after the start button is clicked. It shows that the current frame has no page yet because the simulation has just started. Meanwhile, on the added page it shows the page [5] which is the first page of our reference string and on the new frame it showed [5] in a blue frame because as the legend indicates it is an added frame. We can also see that it says that it was a miss and with that the page fault counter increased to 1.

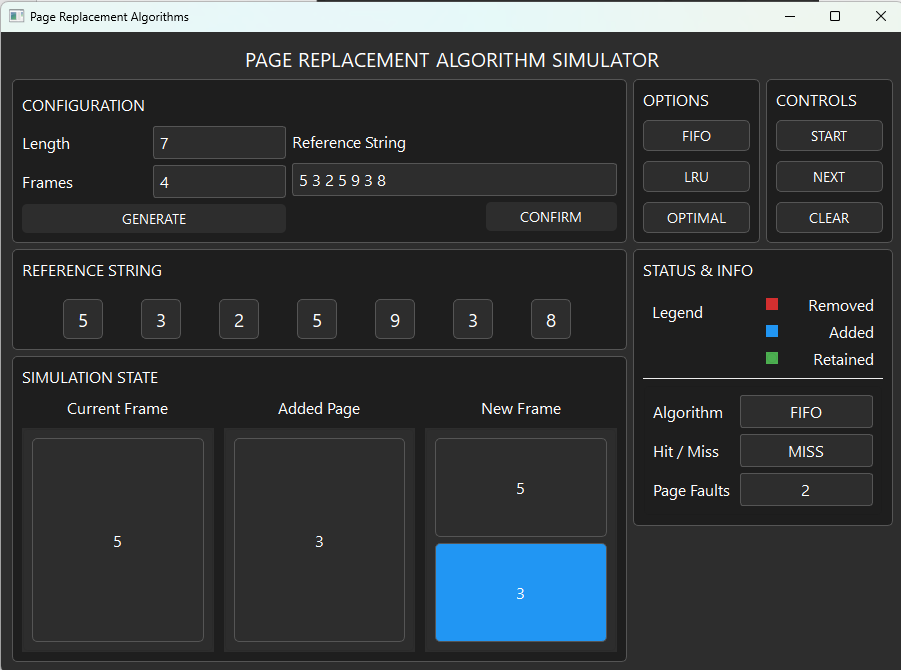


Figure 6.: The second step on the FIFO Simulation 2

In here, the second page reference which is [3] is added. The simulation state section shows the current frame has [5] and the added page is [3] which results to the new frame having [5][3] with [3] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 2.

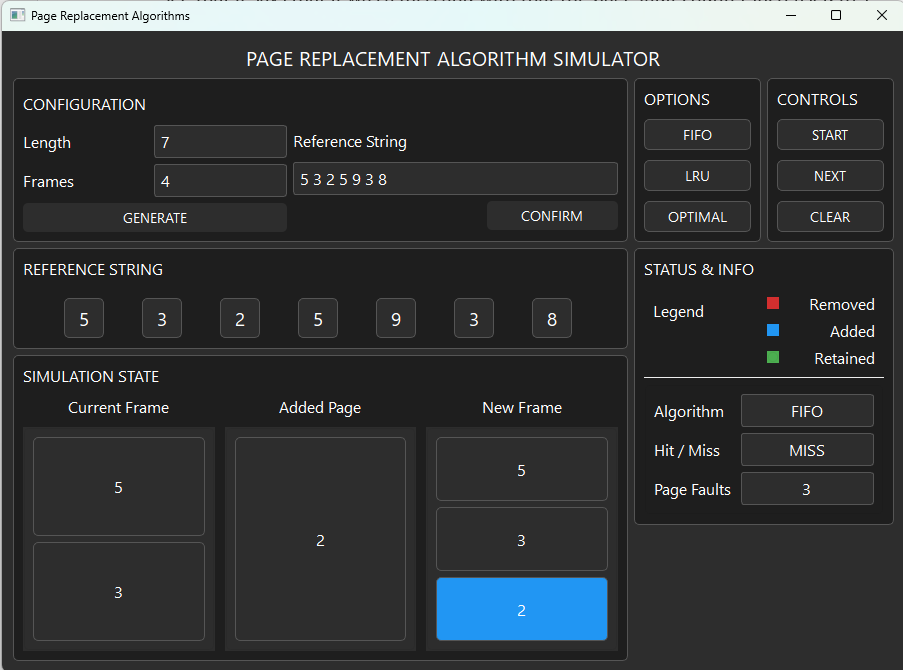


Figure 6.: The third step on the FIFO Simulation 2

In here, the third page reference which is [2] is added. The simulation state section shows the current frame has [5][3] and the added page is [2] which results to the new frame having [5][3][2] with [2] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 3.

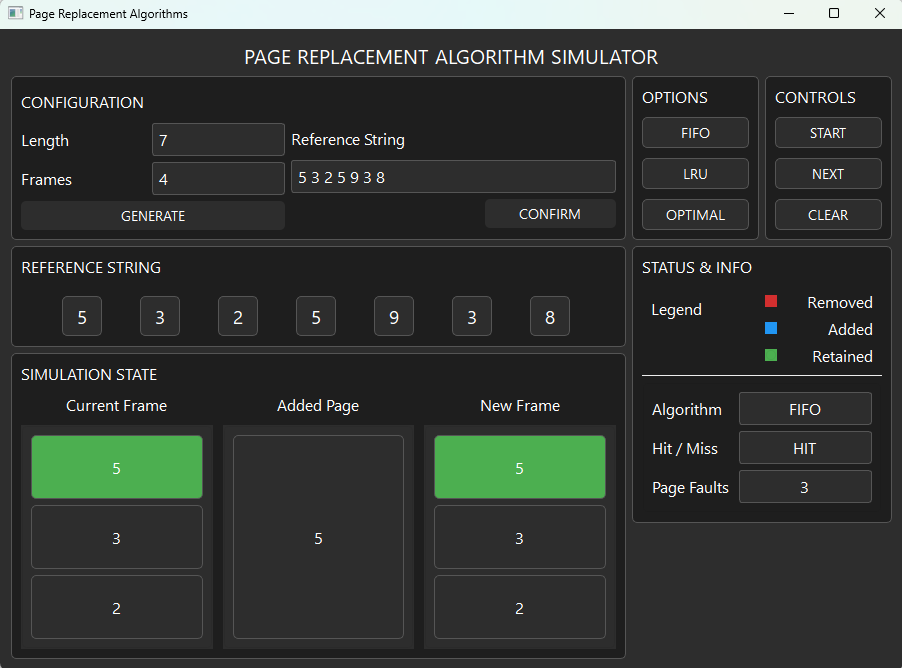


Figure 6.: The fourth step on the FIFO Simulation 2

In here, the fourth page reference which is [5] is being added which is shown on the added page section. It also shows the current frame and the new frame having [5][3][2] with [5] in a green frame because the added page is also a [5] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 3.

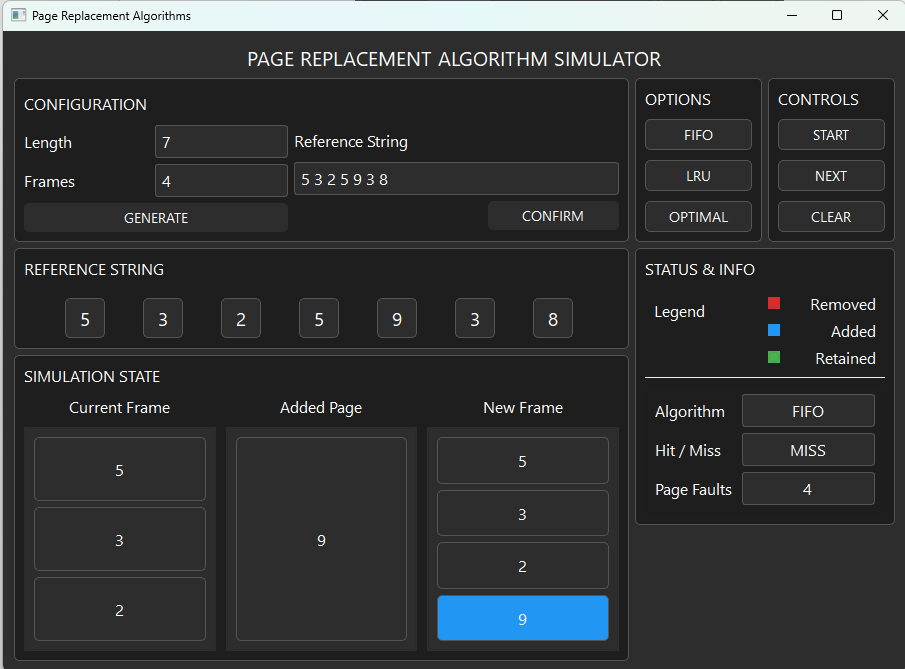


Figure 6.: The fifth step on the FIFO Simulation 2

In here, the fifth page reference which is [9] is added. The simulation state section shows the current frame has [5][3][2] and the added page is [9] which results to the new frame having [5][3][2][9] with [9] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 4.

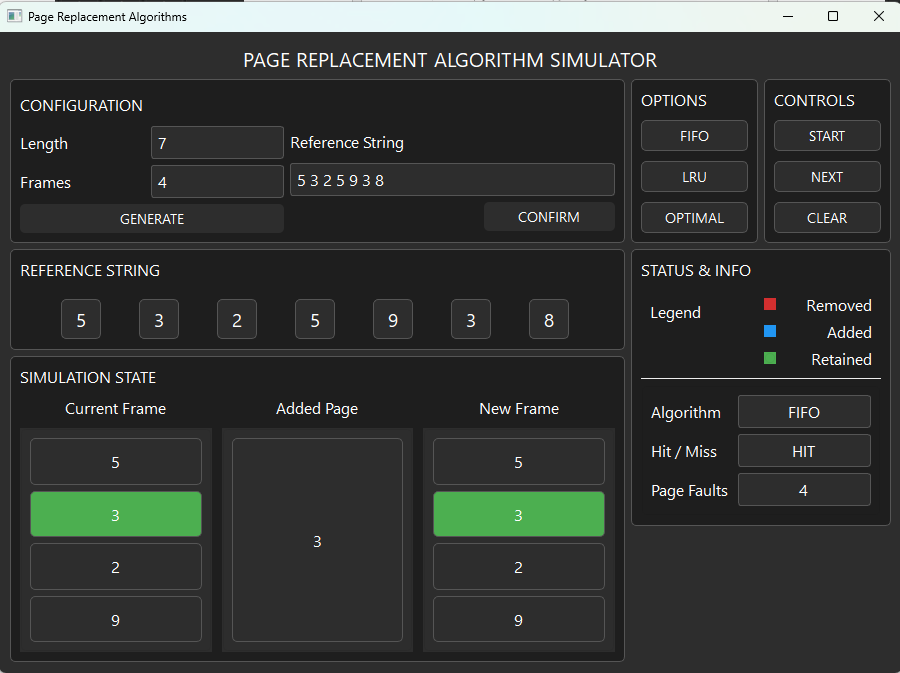


Figure 6.: The sixth step on the FIFO Simulation 2

In here, the sixth page reference which is [3] is being added which is shown on the added page section. It also shows the current frame and the new frame having [5][3][2][9] with [3] in a green frame because the added page is also a [3] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 4.

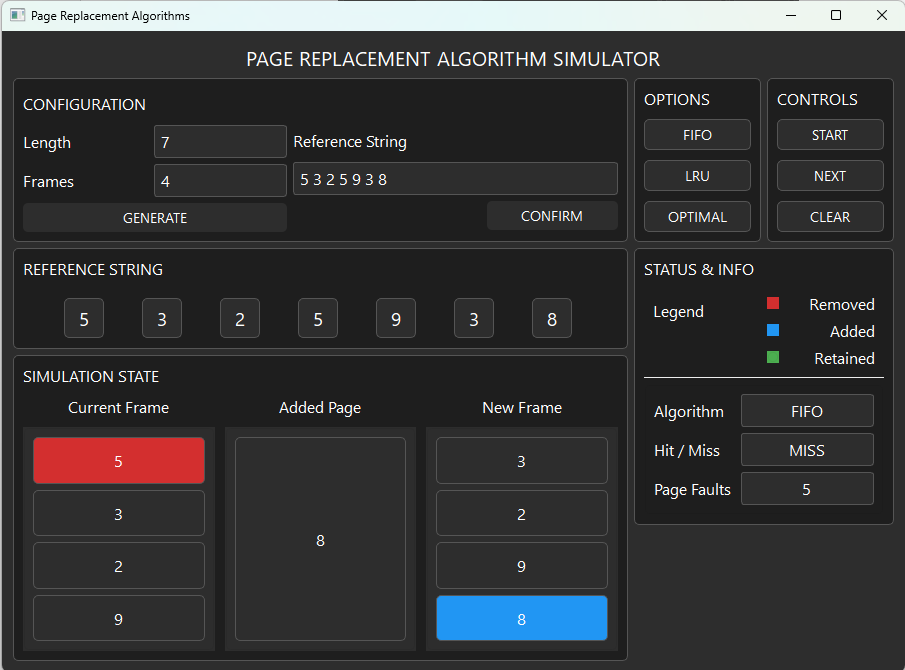


Figure 6.: The seventh step on the FIFO Simulation 2

In here, the seventh page reference which is [8] is added. The simulation state section shows the current frame has [5][3][2][9] with [5] in a red frame because it is being removed and the added page is [8] which results to the new frame having [3][2][9][8] with [8] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 5.

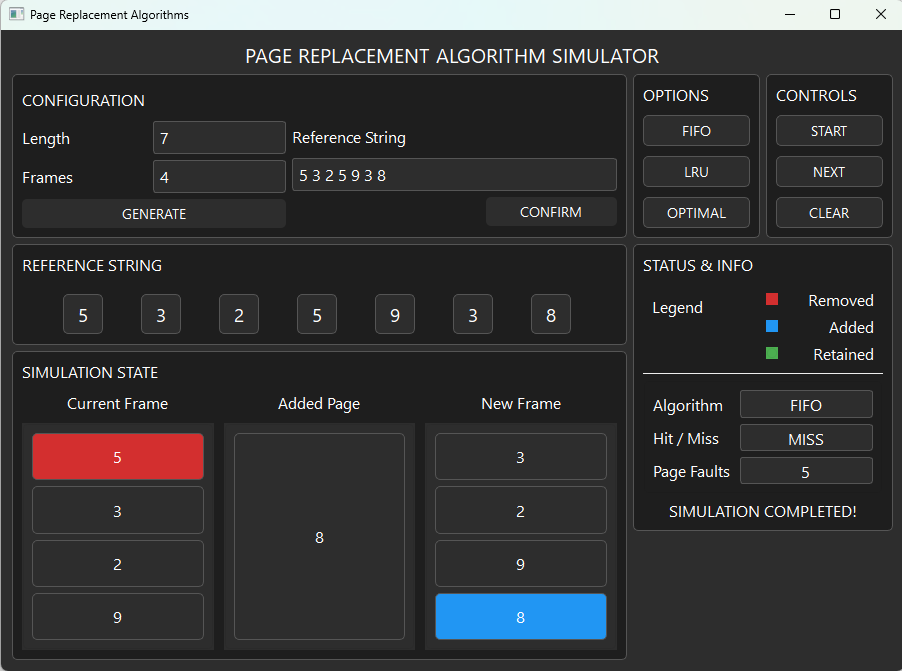


Figure 6.: The eighth step on the FIFO Simulation 2

This shows that the Simulation Completed text has been visible which states that our simulation has finished and the final number of page faults in this simulation is 5.

### LRU Algorithm Outputs

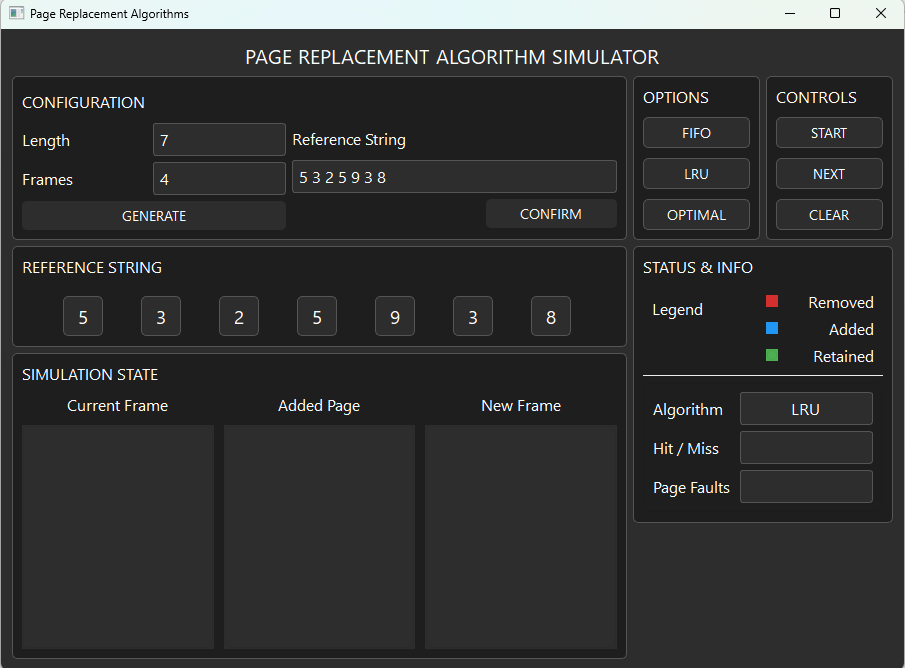


Figure 7.1: Initial Output for LRU Simulation 2

This shows the initial output before the start button has been clicked. As it can be seen in the image the provided length and frames are 7 and 4 respectively. The reference string generated is also shown on the reference string section and the LRU algorithm is also shown on the status & info section.

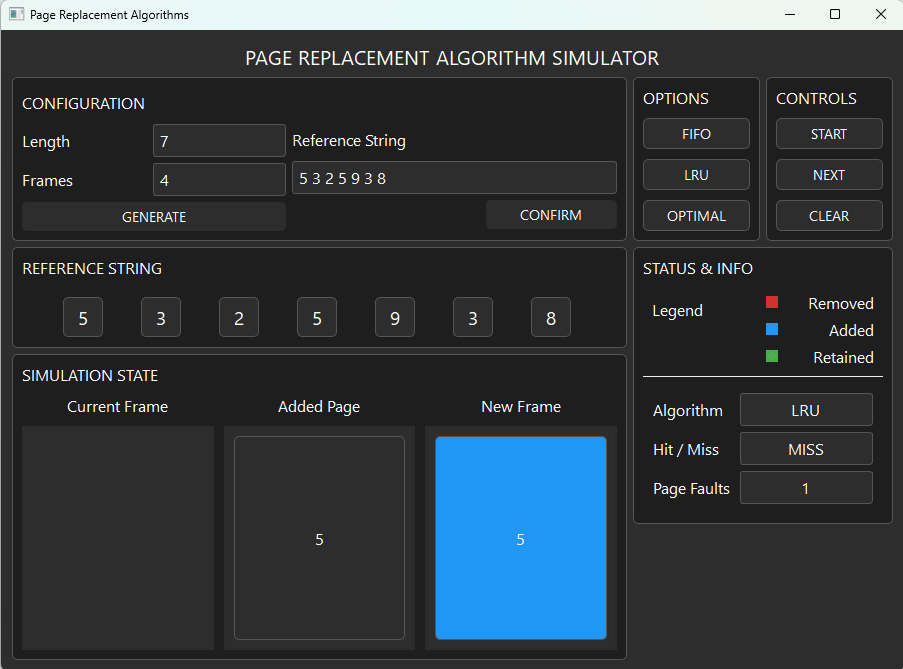


Figure 7.2: Output of LRU Simulation 2 after the Start Button is Pressed

This shows the output after the start button is clicked. It shows that the current frame has no page yet because the simulation has just started. Meanwhile, on the added page it shows the page [5] which is the first page of our reference string and on the new frame it showed [5] in a blue frame because as the legend indicates it is an added frame. We can also see that it says that it was a miss and with that the page fault counter increased to 1.

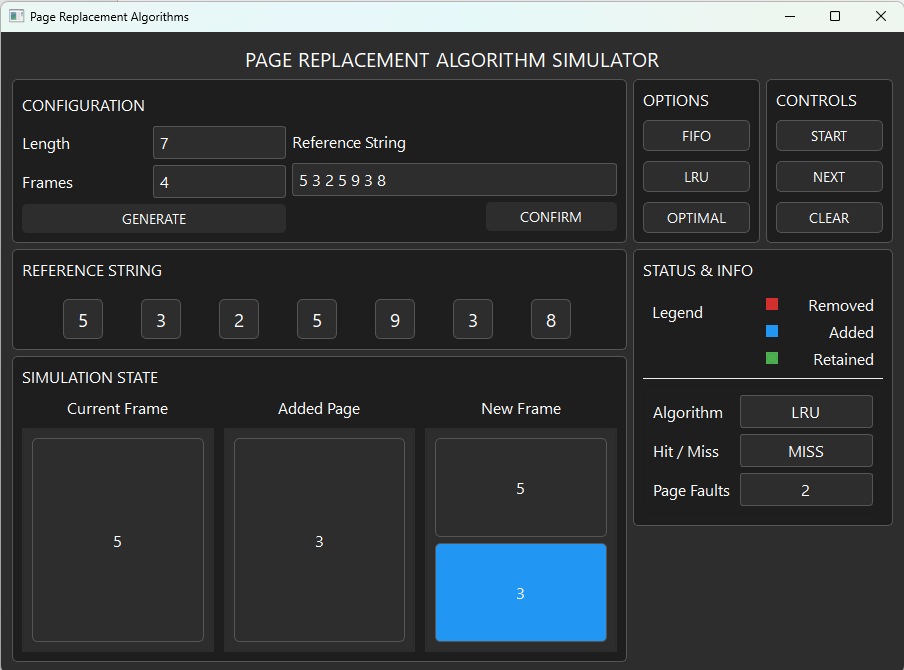


Figure 7.3: The second step on the LRU Simulation 2

In here, the second page reference which is [3] is added. The simulation state section shows the current frame has [5] and the added page is [3] which results to the new frame having [5][3] with [3] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 2.

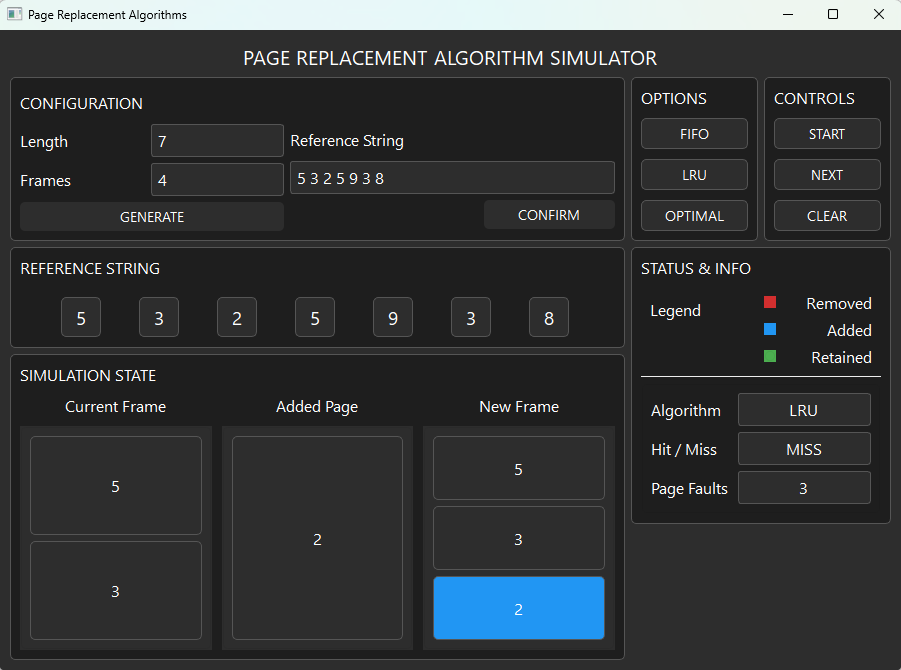


Figure 7.4: The third step on the LRU Simulation 2

In here, the third page reference which is [2] is added. The simulation state section shows the current frame has [5][3] and the added page is [2] which results to the new frame having [5][3][2] with [2] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 3.

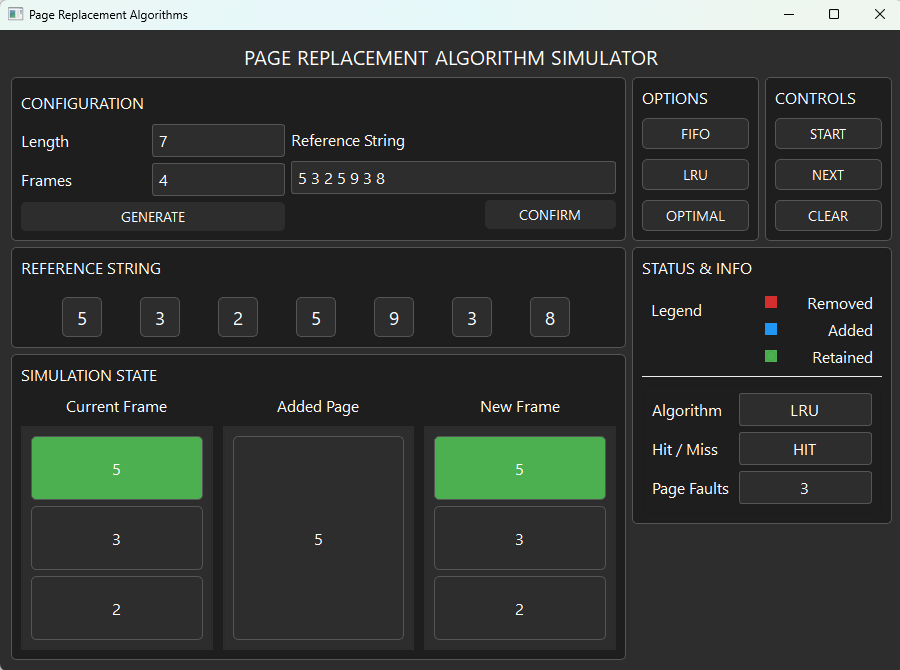


Figure 7.5: The fourth step on the LRU Simulation 2

In here, the fourth page reference which is [5] is being added which is shown on the added page section. It also shows the current frame and the new frame having [5][3][2] with [5] in a green frame because the added page is also a [5] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 3.

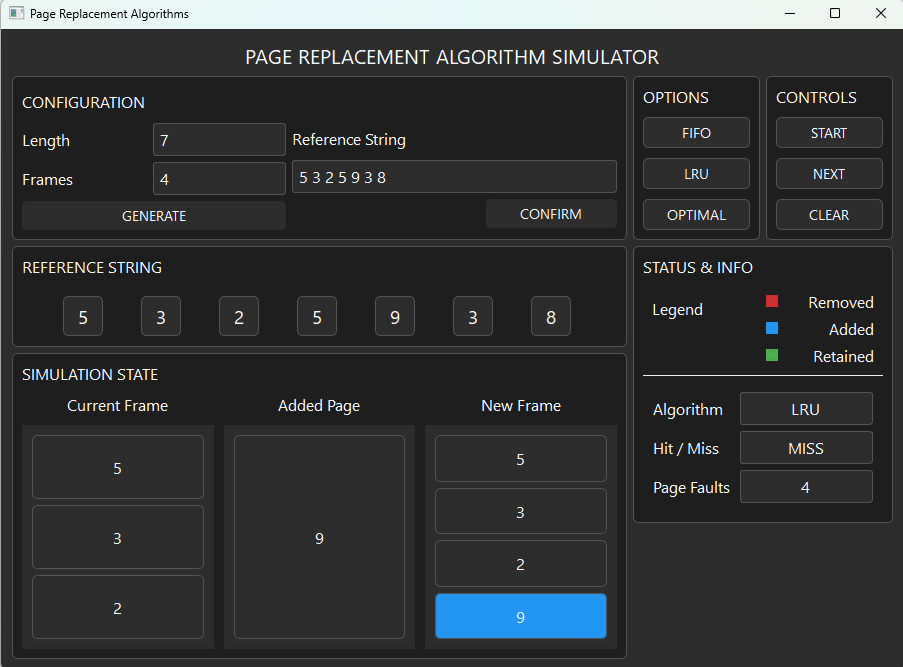


Figure 7.6: The fifth step on the LRU Simulation 2

In here, the fifth page reference which is [9] is added. The simulation state section shows the current frame has [5][3][2] and the added page is [9] which results to the new frame having [5][3][2][9] with [9] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 4.

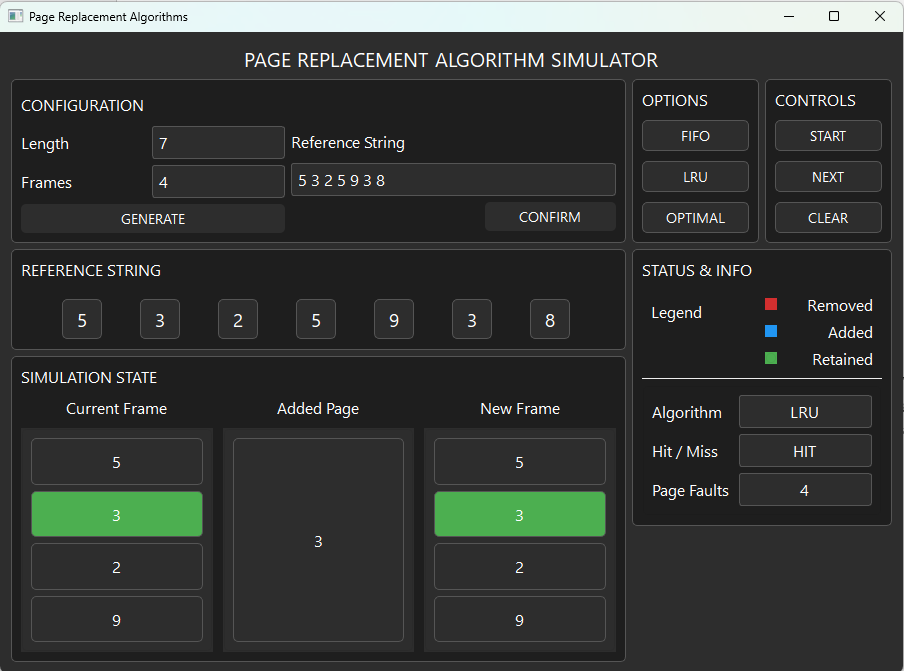


Figure 7.7: The sixth step on the LRU Simulation 2

In here, the sixth page reference which is [3] is being added which is shown on the added page section. It also shows the current frame and the new frame having [5][3][2][9] with [3] in a green frame because the added page is also a [3] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 4.

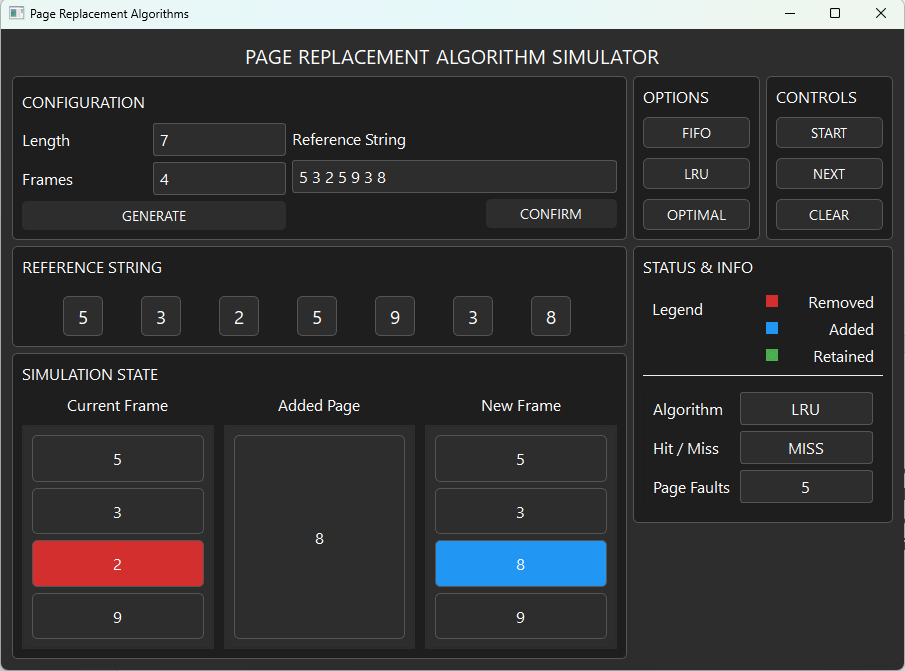


Figure 7.8: The seventh step on the LRU Simulation 2

In here, the seventh page reference which is [8] is added. The simulation state section shows the current frame has [5][3][2][9] with [2] in a red frame because it is being removed and the added page is [8] which results to the new frame having [5][3][8][9] with [8] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 5.

### Optimal Algorithm Outputs

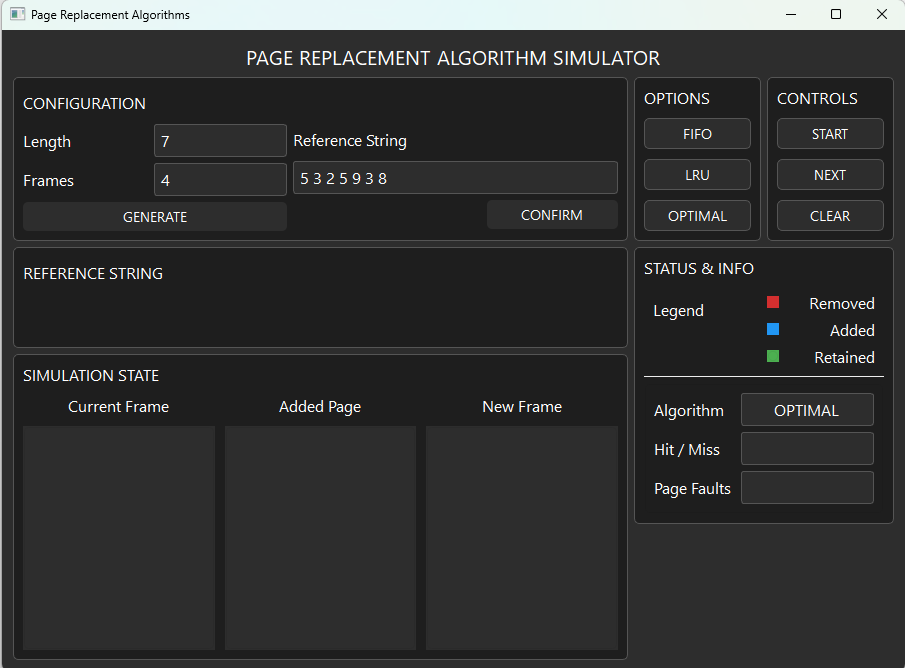


Figure 8.: Initial Output for Optimal Simulation 2

This shows the initial output before the start button has been clicked. As it can be seen in the image the provided length and frames are 7 and 4 respectively. The reference string generated is also shown on the reference string section and the Optimal algorithm is also shown on the status & info section.

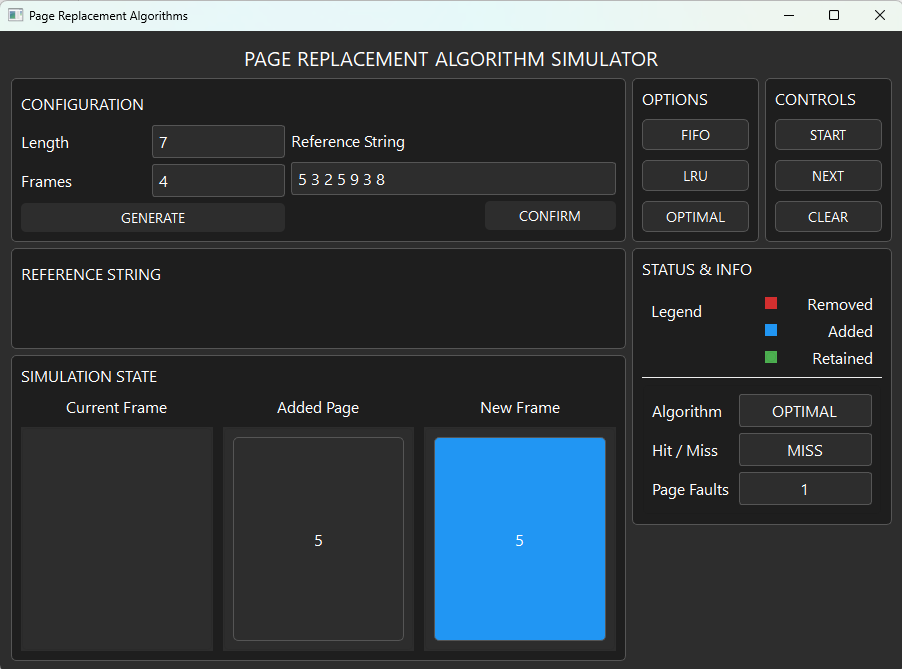


Figure 8.: Output of Optimal Simulation 2 after the Start Button is Pressed

This shows the output after the start button is clicked. It shows that the current frame has no page yet because the simulation has just started. Meanwhile, on the added page it shows the page [5] which is the first page of our reference string and on the new frame it showed [5] in a blue frame because as the legend indicates it is an added frame. We can also see that it says that it was a miss and with that the page fault counter increased to 1.

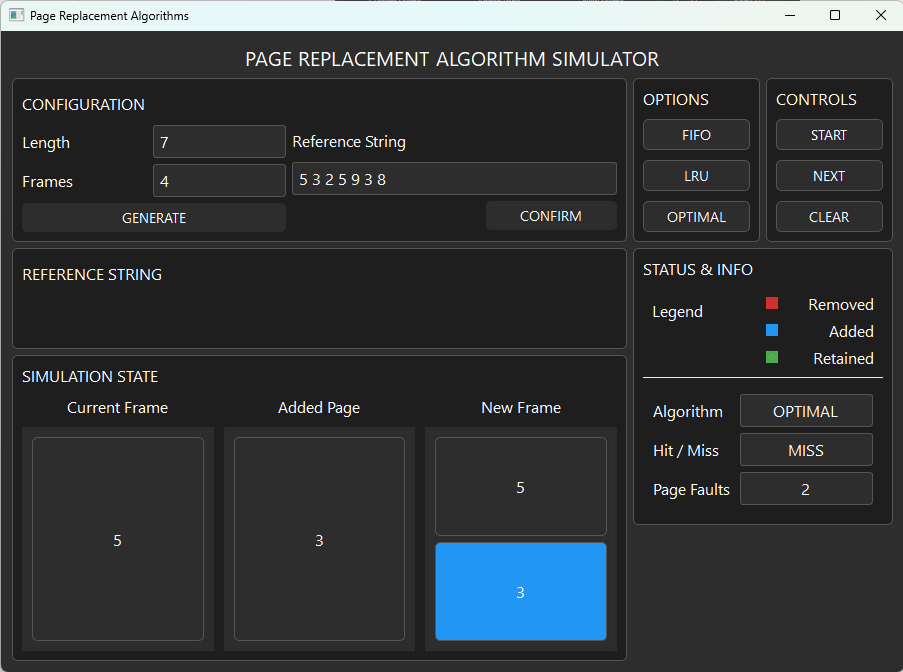


Figure 8.: The second step on the Optimal Simulation 2

In here, the second page reference which is [3] is added. The simulation state section shows the current frame has [5] and the added page is [3] which results to the new frame having [5][3] with [3] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 2.

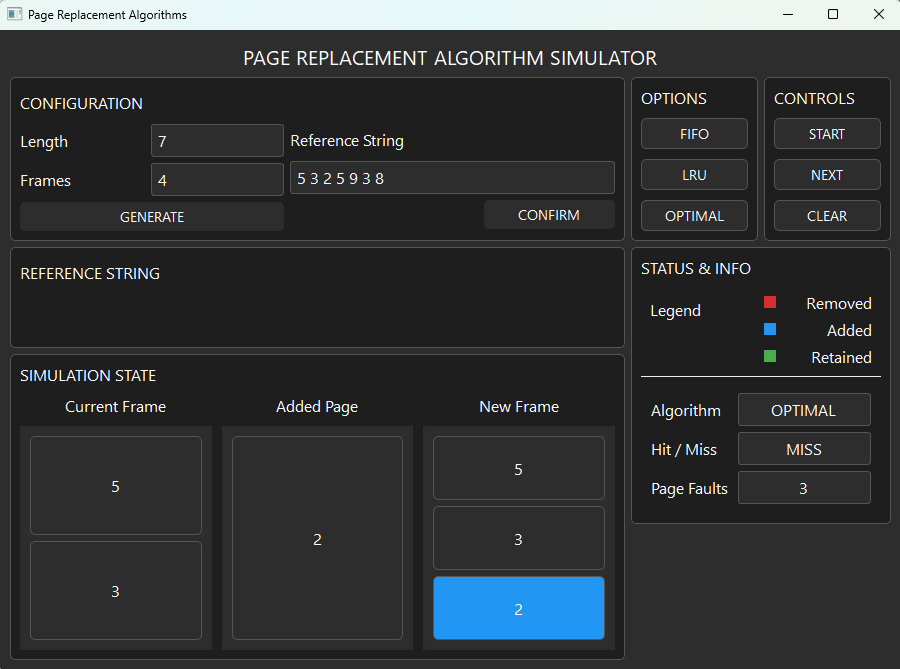


Figure 8.: The third step on the Optimal Simulation 2

In here, the third page reference which is [2] is added. The simulation state section shows the current frame has [5][3] and the added page is [2] which results to the new frame having [5][3][2] with [2] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 3.

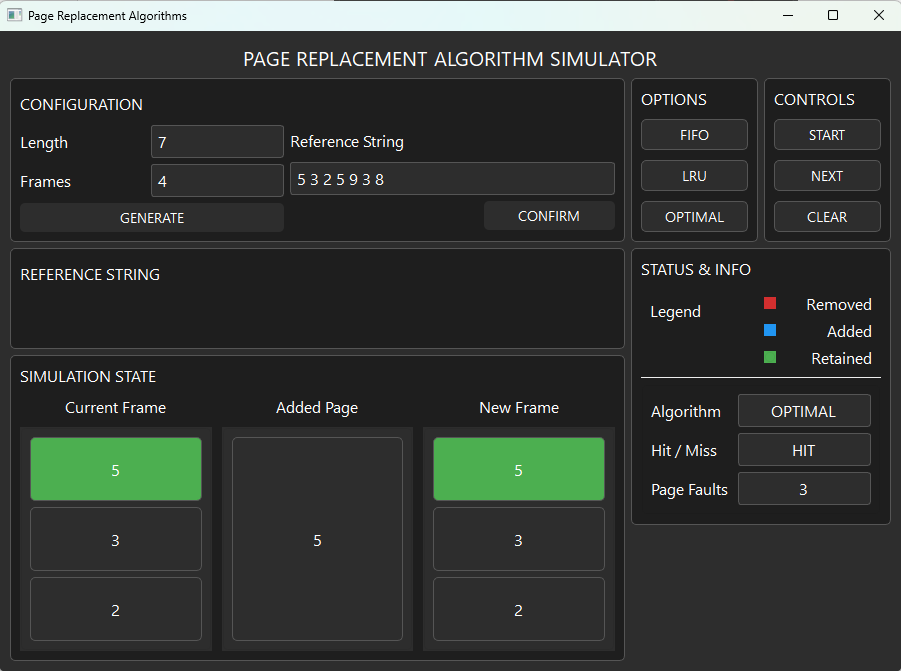


Figure 8.: The fourth step on the Optimal Simulation 2

In here, the fourth page reference which is [5] is being added which is shown on the added page section. It also shows the current frame and the new frame having [5][3][2] with [5] in a green frame because the added page is also a [5] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 3.

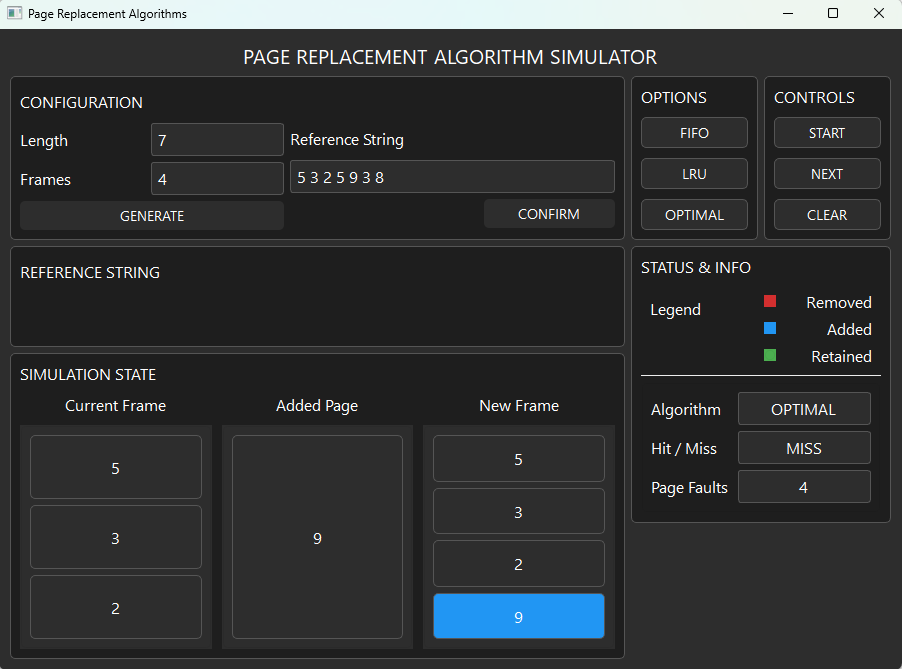


Figure 8.: The fifth step on the Optimal Simulation 2

In here, the fifth page reference which is [9] is added. The simulation state section shows the current frame has [5][3][2] and the added page is [9] which results to the new frame having [5][3][2][9] with [9] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 4.

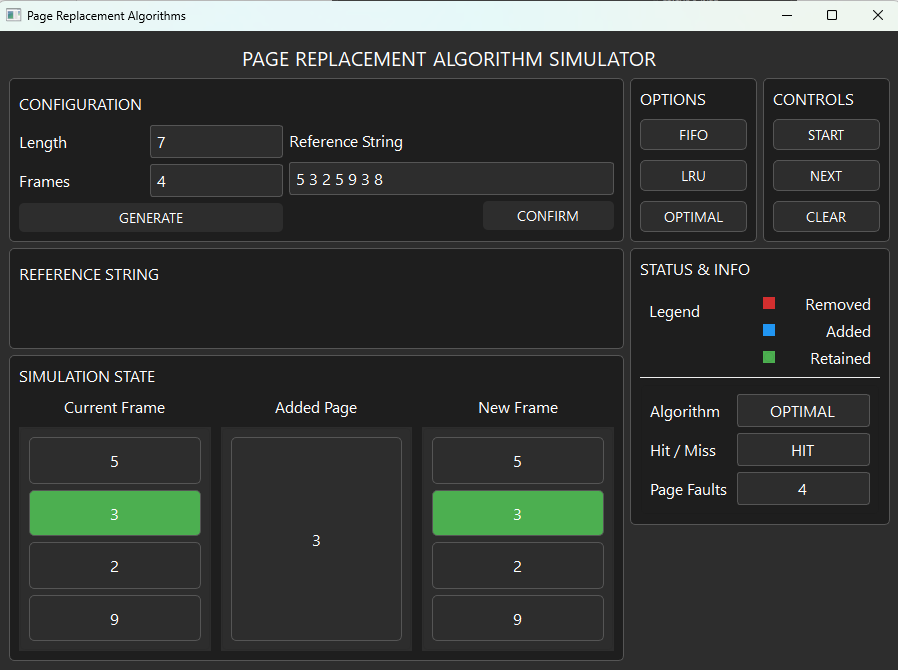


Figure 8.: The sixth step on the Optimal Simulation 2

In here, the sixth page reference which is [3] is being added which is shown on the added page section. It also shows the current frame and the new frame having [5][3][2][9] with [3] in a green frame because the added page is also a [3] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 4.

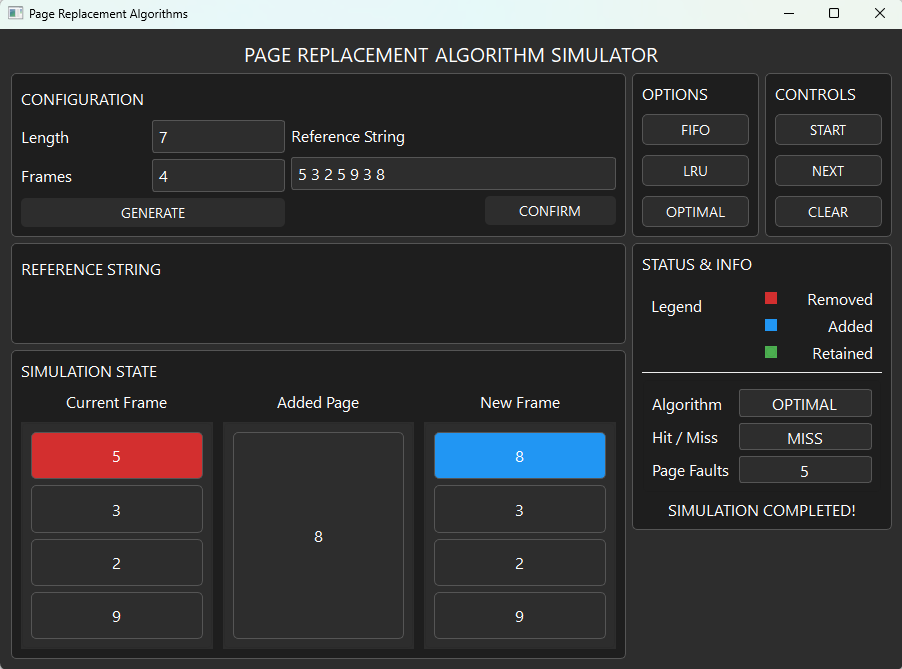


Figure 8.: The seventh step on the Optimal Simulation 2

In here, the seventh page reference which is [8] is added. The simulation state section shows the current frame has [5][3][2][9] with [5] in a red frame because it is being removed and the added page is [8] which results to the new frame having [8][3][2][9] with [8] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 5. It also shows that the Simulation Completed text has been visible which states that our simulation has finished and the final number of page faults in this simulation is 5.

## Simulation 3 [ 8, 0, 1, 8, 3 ]

### FIFO Algorithm Outputs

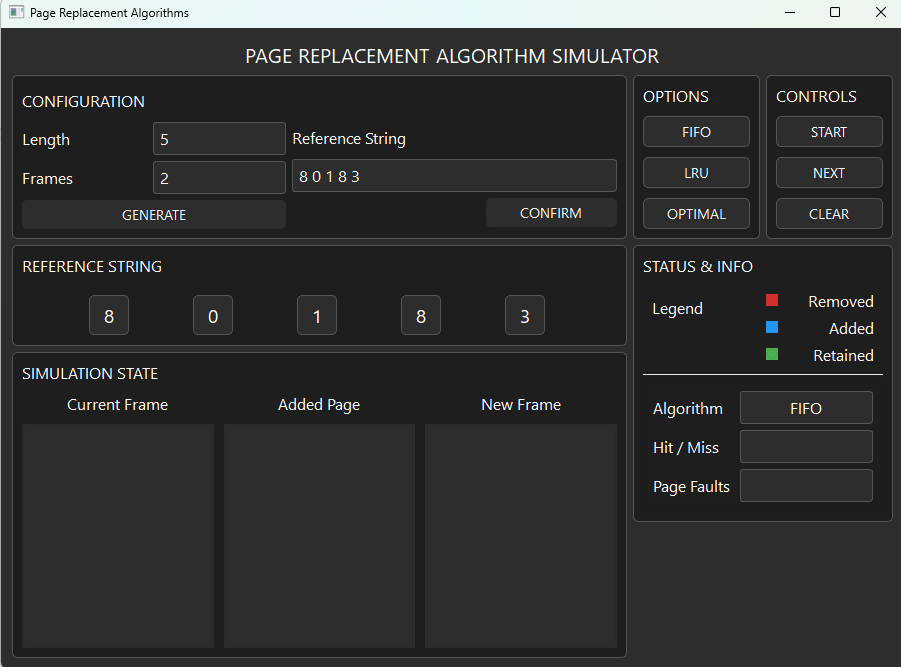


Figure 9.: Initial Output for FIFO Simulation 3

This shows the initial output before the start button has been clicked. As it can be seen in the image the provided length and frames are 5 and 2 respectively. The reference string generated is also shown on the reference string section and the FIFO algorithm is also shown on the status & info section.

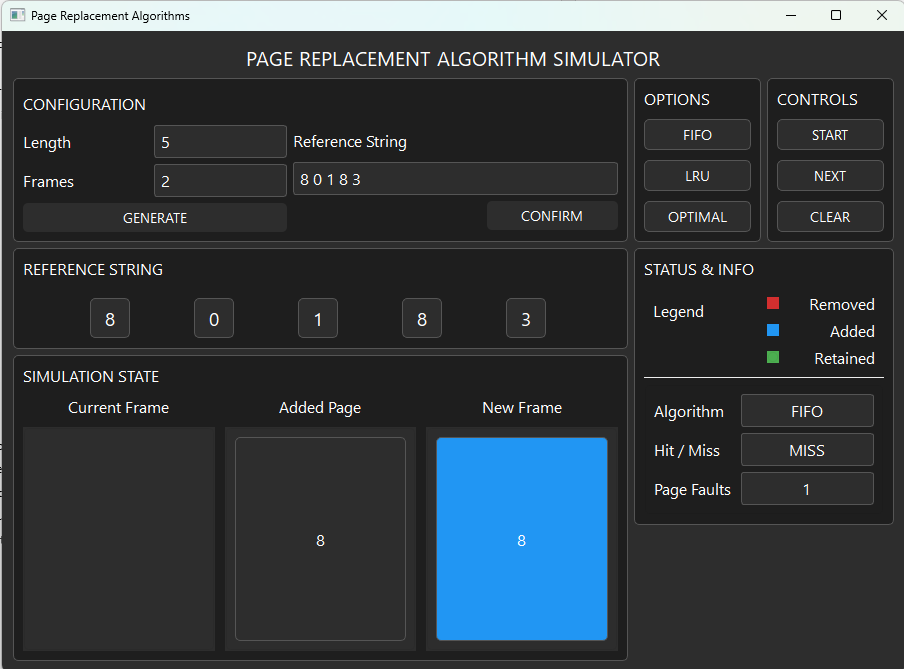


Figure 9.: Output of FIFO Simulation 3 after the Start Button is Pressed

This shows the output after the start button is clicked. It shows that the current frame has no page yet because the simulation has just started. Meanwhile, on the added page it shows the page [8] which is the first page of our reference string and on the new frame it showed [8] in a blue frame because as the legend indicates it is an added frame. We can also see that it says that it was a miss and with that the page fault counter increased to 1.

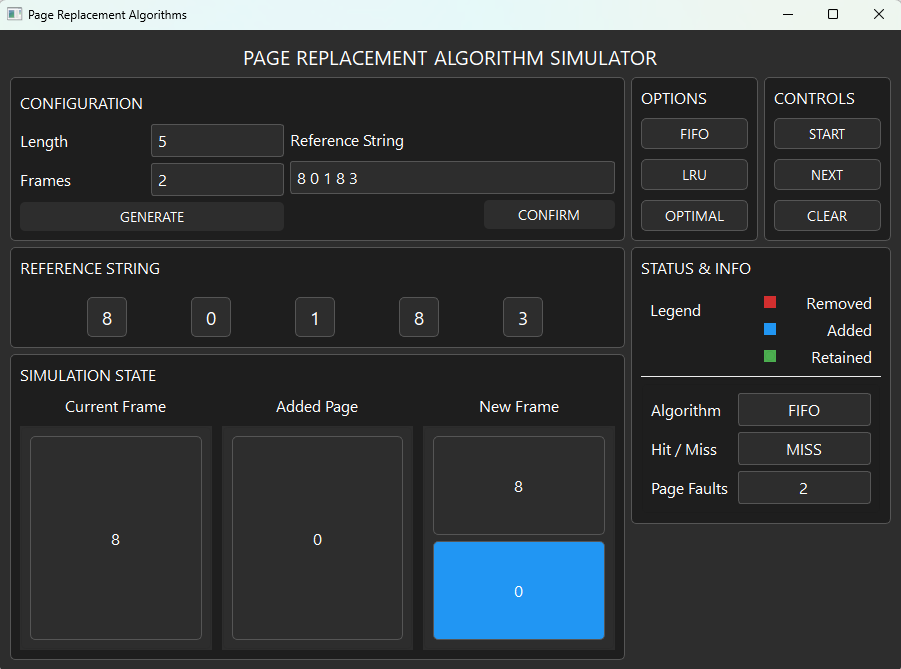


Figure 9.: The second step on the FIFO Simulation 3

In here, the second page reference which is [0] is added. The simulation state section shows the current frame has [8] and the added page is [0] which results to the new frame having [8][0] with [0] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 2.

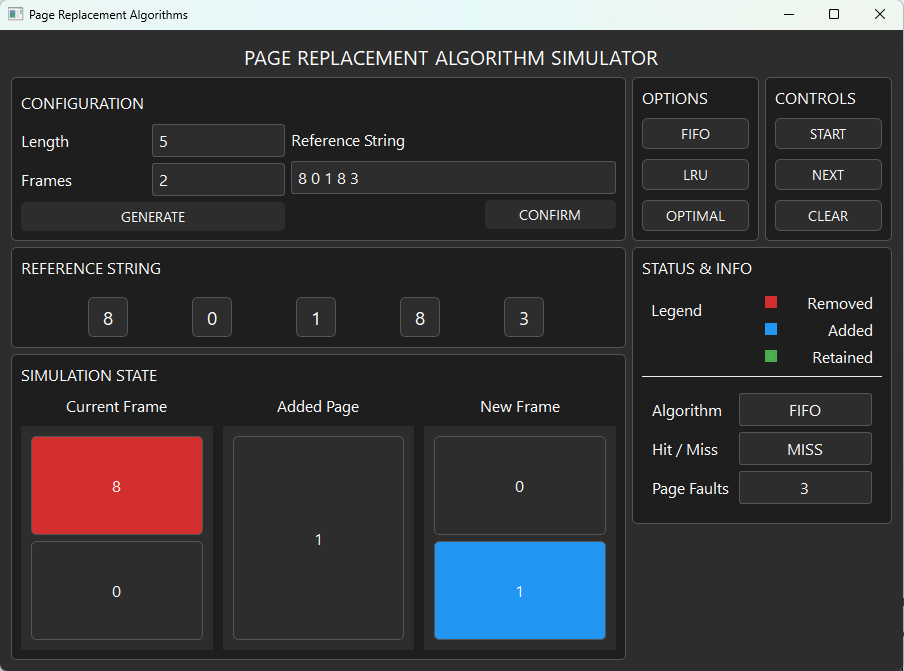


Figure 9.: The third step on the FIFO Simulation 3

In here, the third page reference which is [1] is added. The simulation state section shows the current frame has [8][0] with [8] in a red frame because it is being removed and the added page is [1] which results to the new frame having [0][1] with [1] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 3.

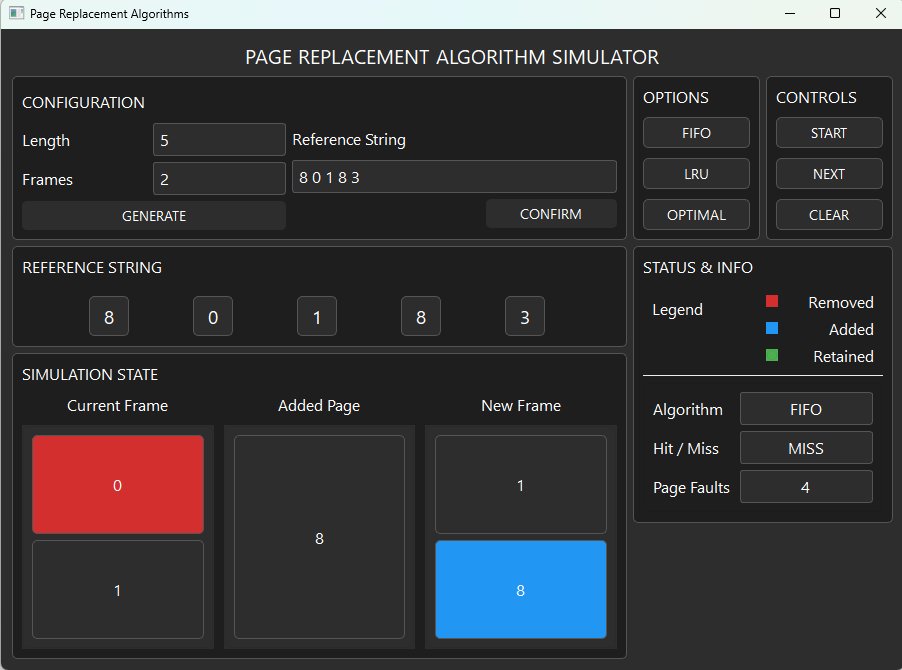


Figure 9.: The fourth step on the FIFO Simulation 3

In here, the fourth page reference which is [8] is added. The simulation state section shows the current frame has [0][1] with [0] in a red frame because it is being removed and the added page is [8] which results to the new frame having [1][8] with [8] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 4.

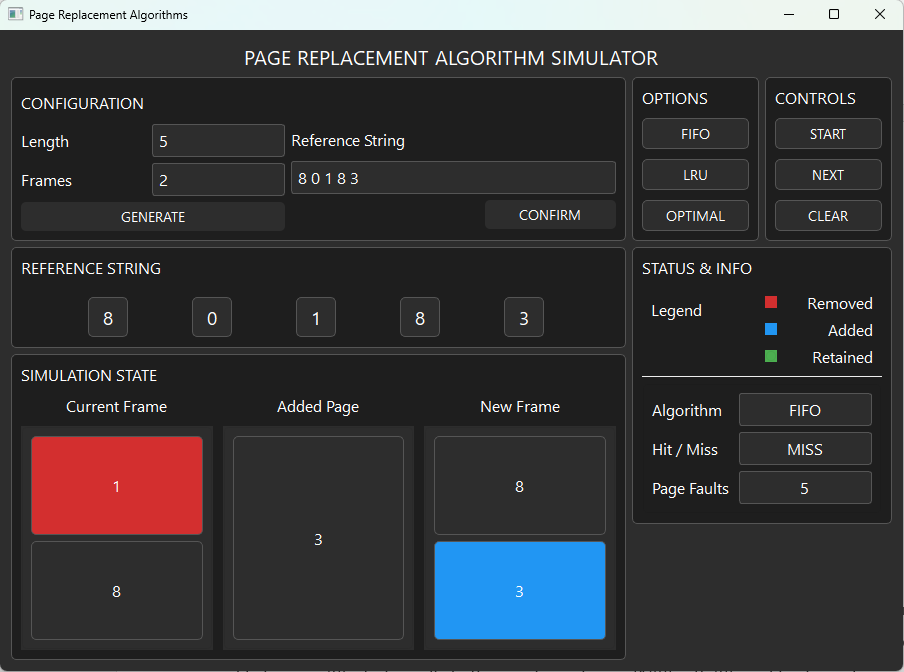


Figure 9.: The fifth step on the FIFO Simulation 3

In here, the fifth page reference which is [3] is added. The simulation state section shows the current frame has [1][8] with [1] in a red frame because it is being removed and the added page is [3] which results to the new frame having [8][3] with [3] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 5.

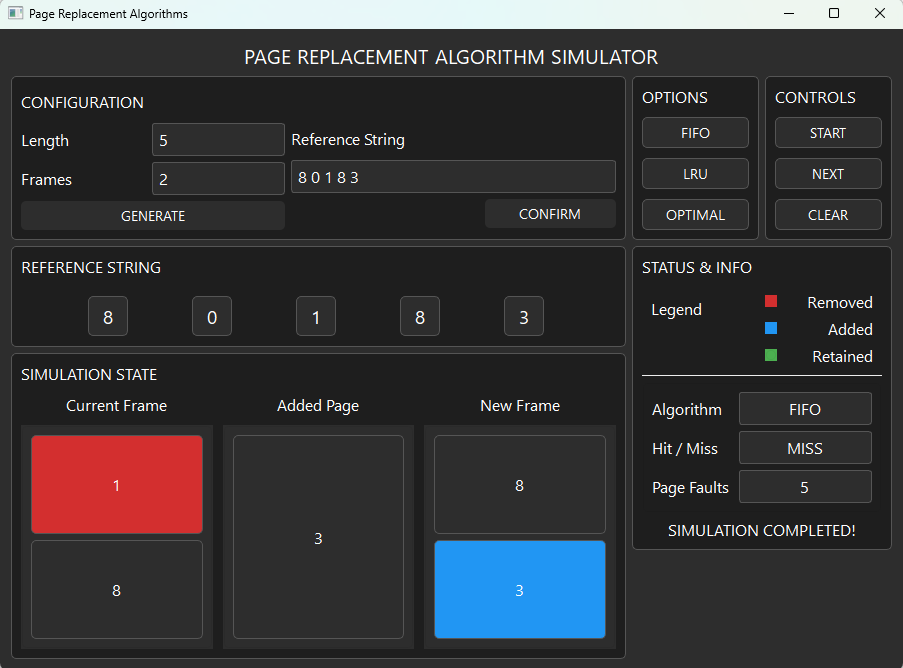


Figure 9.: The sixth step on the FIFO Simulation 3

This shows that the Simulation Completed text has been visible which states that our simulation has finished and the final number of page faults in this simulation is 5.

### LRU Algorithm Outputs

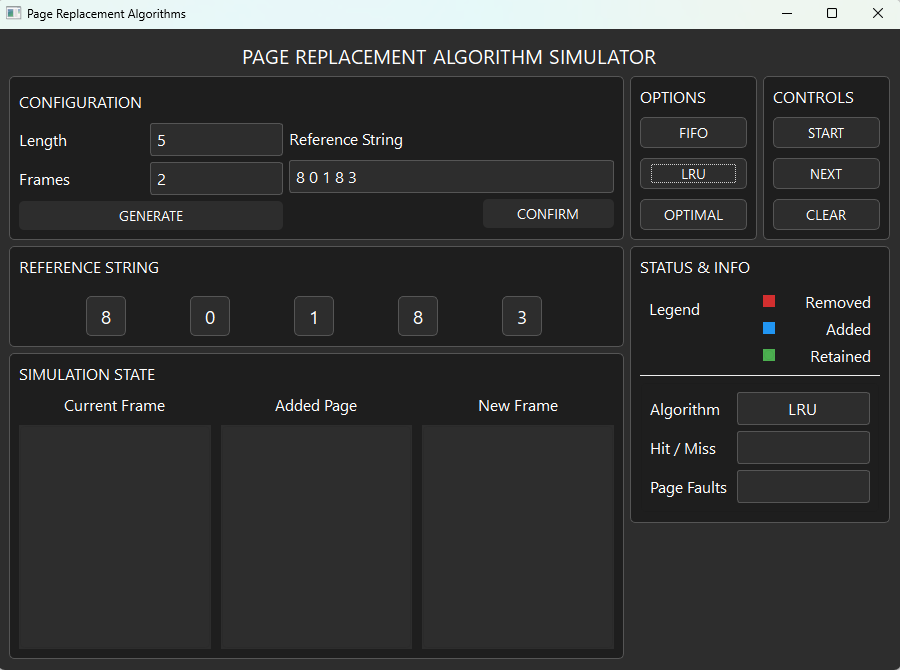


Figure 10.: Initial Output for LRU Simulation 3

This shows the initial output before the start button has been clicked. As it can be seen in the image the provided length and frames are 5 and 2 respectively. The reference string generated is also shown on the reference string section and the LRU algorithm is also shown on the status & info section.

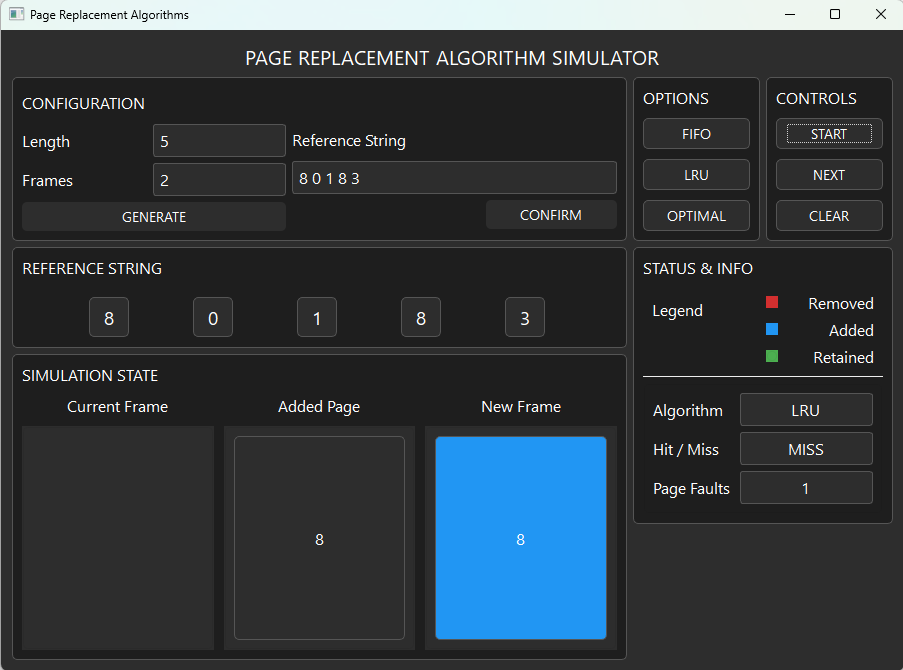


Figure 10.: Output of LRU Simulation 3 after the Start Button is Pressed

This shows the output after the start button is clicked. It shows that the current frame has no page yet because the simulation has just started. Meanwhile, on the added page it shows the page [8] which is the first page of our reference string and on the new frame it showed [8] in a blue frame because as the legend indicates it is an added frame. We can also see that it says that it was a miss and with that the page fault counter increased to 1.

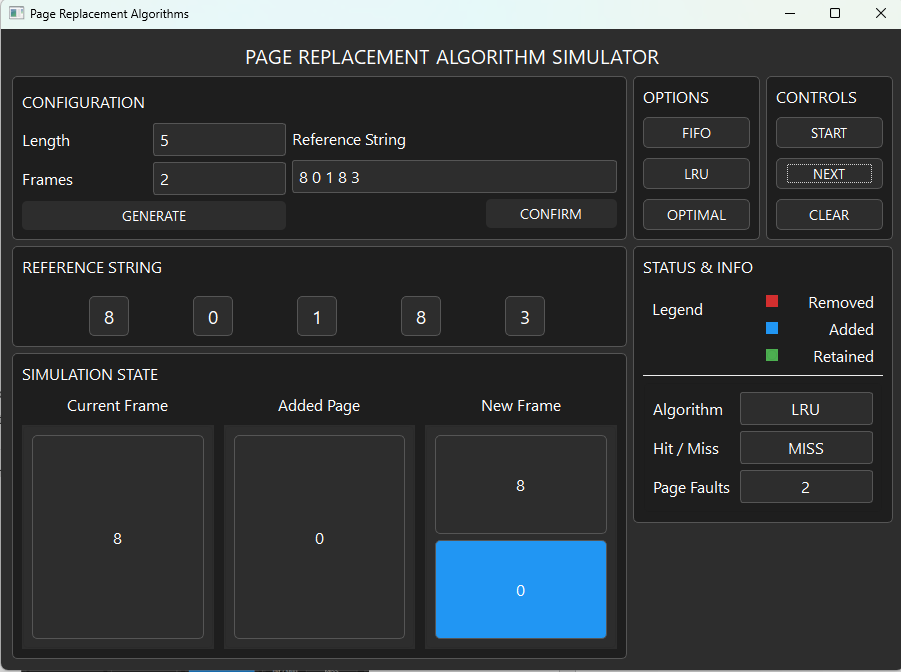


Figure 10.: The second step on the LRU Simulation 3

In here, the second page reference which is [0] is added. The simulation state section shows the current frame has [8] and the added page is [0] which results to the new frame having [8][0] with [0] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 2.

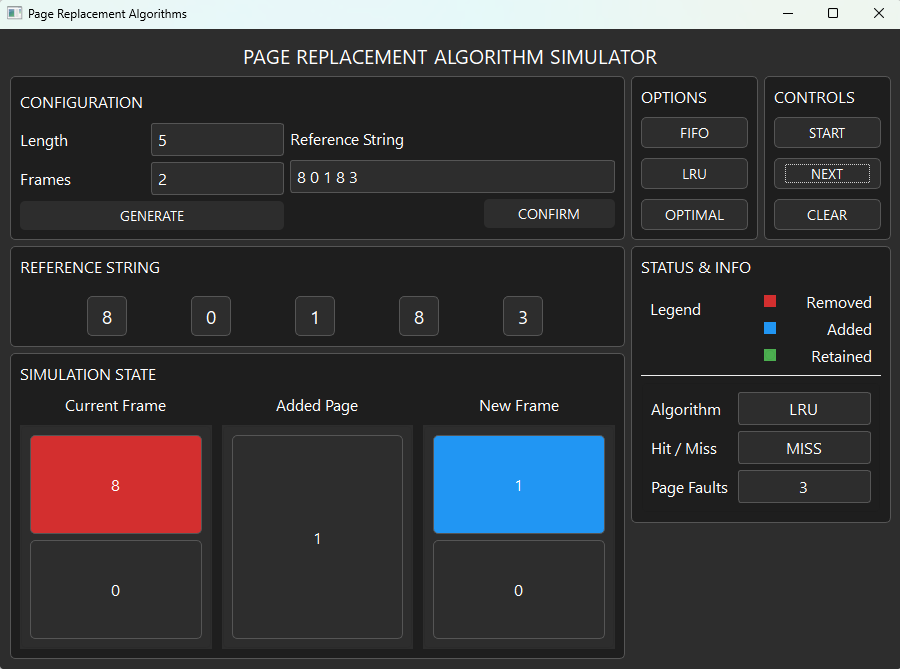


Figure 10.: The third step on the LRU Simulation 3

In here, the third page reference which is [1] is added. The simulation state section shows the current frame has [8][0] with [8] in a red frame because it is being removed and the added page is [1] which results to the new frame having [1][0] with [1] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 3.

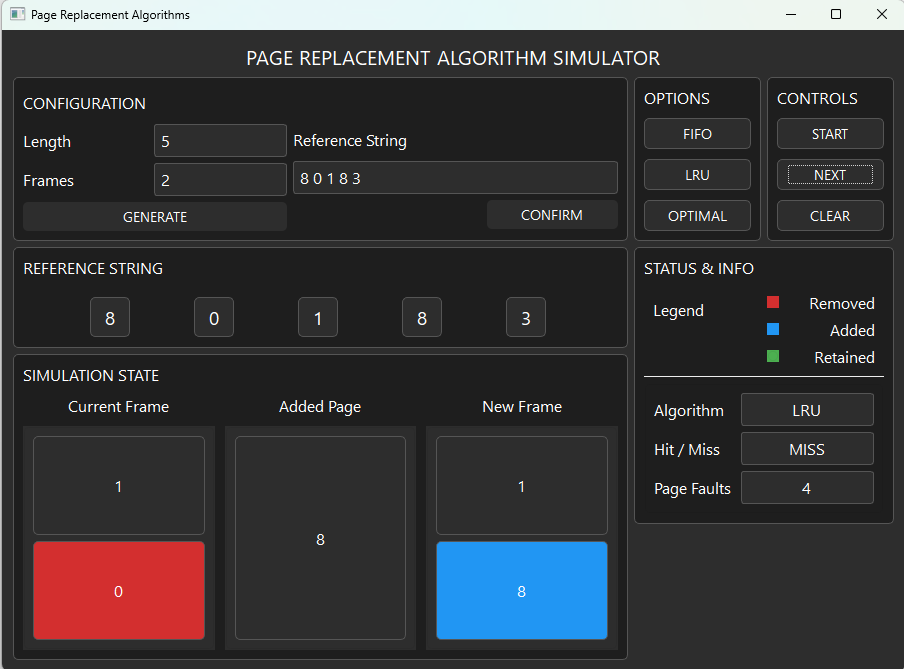


Figure 10.: The fourth step on the LRU Simulation 3

In here, the fourth page reference which is [8] is added. The simulation state section shows the current frame has [1][0] with [0] in a red frame because it is being removed and the added page is [8] which results to the new frame having [1][8] with [8] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 4.

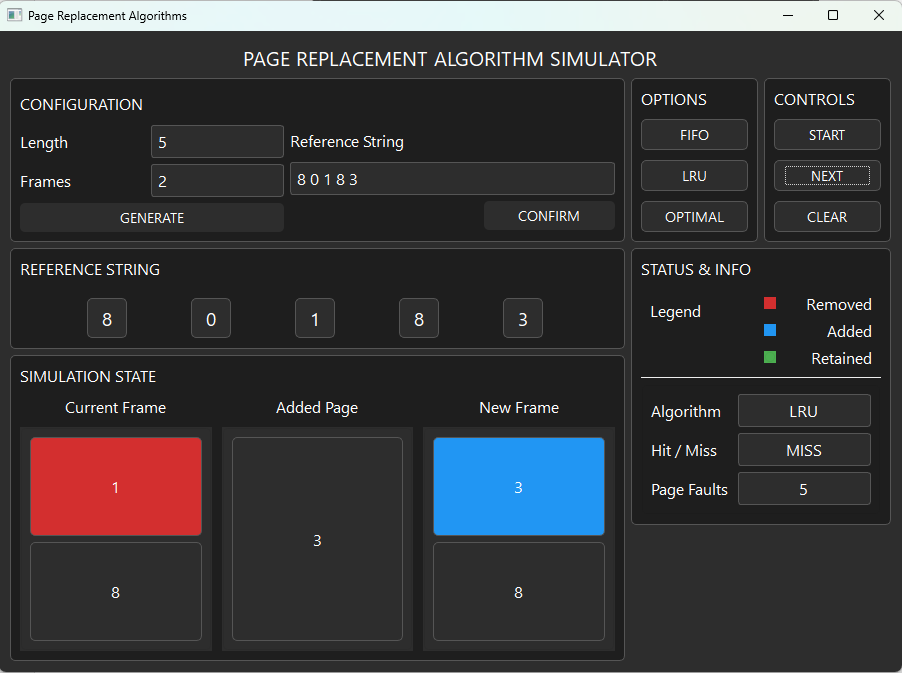


Figure 10.: The fifth step on the LRU Simulation 3

In here, the fifth page reference which is [3] is added. The simulation state section shows the current frame has [1][8] with [1] in a red frame because it is being removed and the added page is [3] which results to the new frame having [3][8] with [3] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 5.

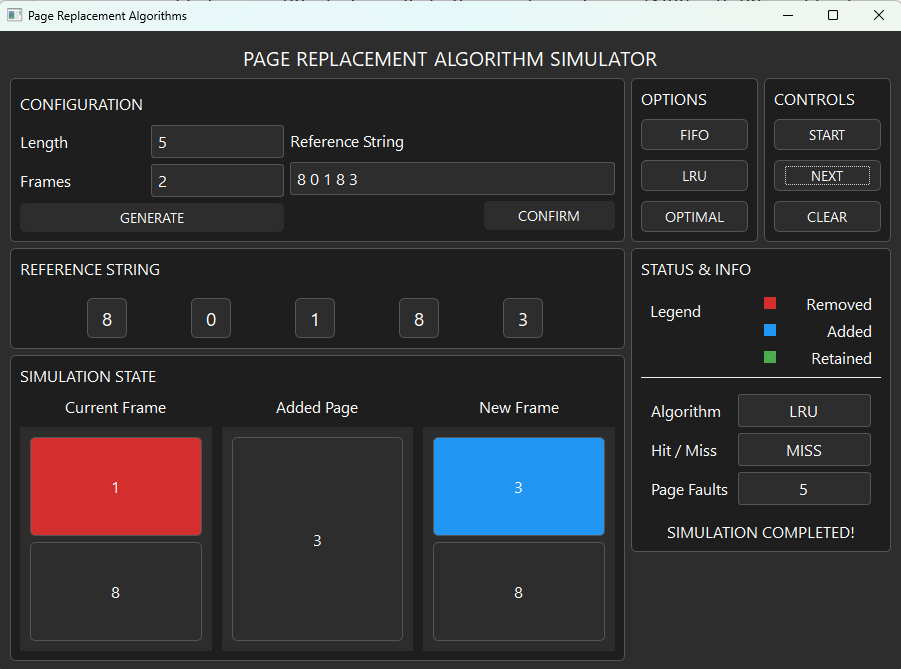


Figure 10.: The sixth step on the LRU Simulation 3

This shows that the Simulation Completed text has been visible which states that our simulation has finished and the final number of page faults in this simulation is 5.

### Optimal Algorithm Outputs

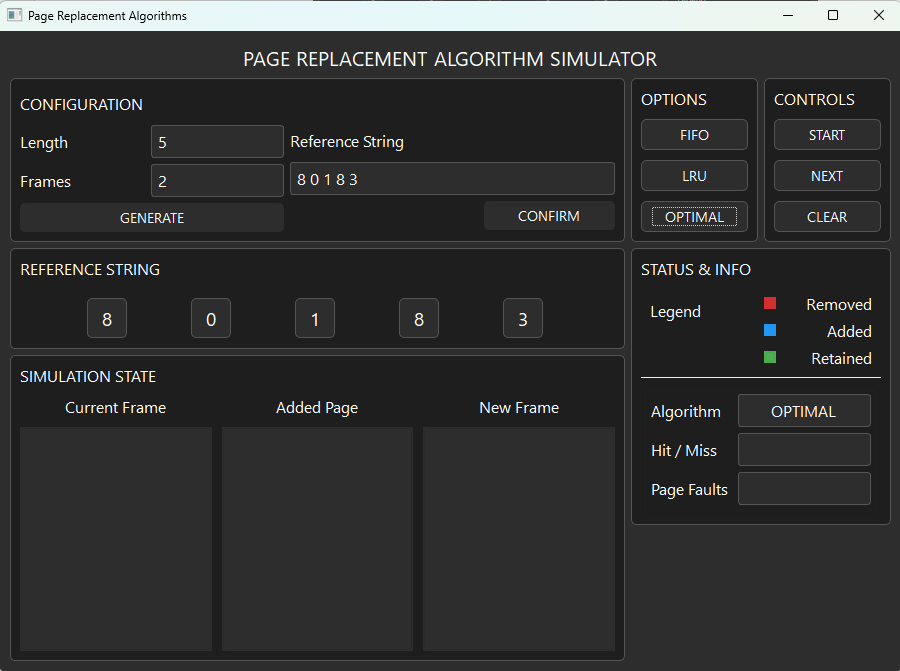


Figure 11.: Initial Output for Optimal Simulation 3

This shows the initial output before the start button has been clicked. As it can be seen in the image the provided length and frames are 5 and 2 respectively. The reference string generated is also shown on the reference string section and the Optimal algorithm is also shown on the status & info section.

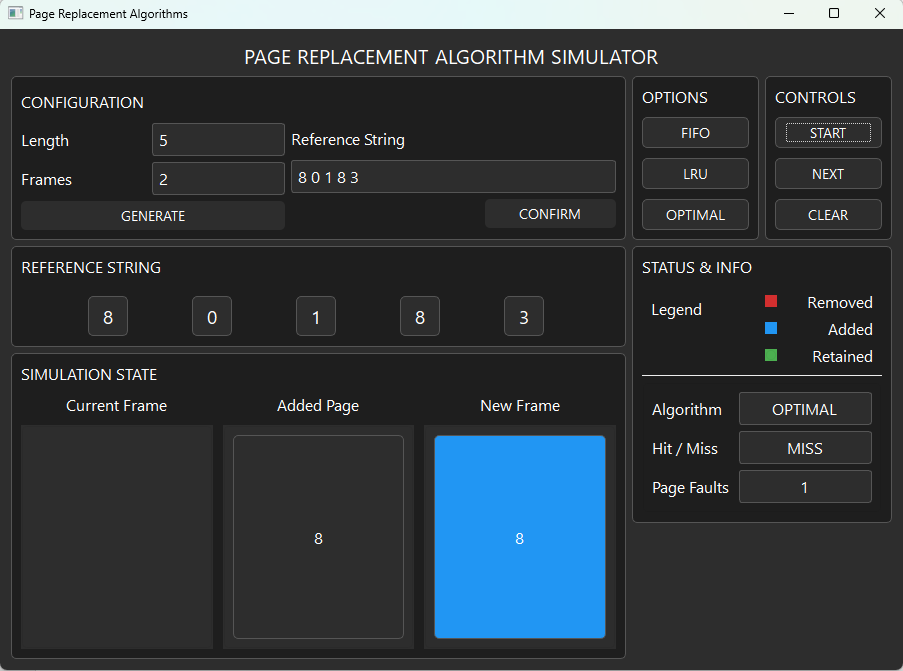


Figure 11.: Output of Optimal Simulation 3 after the Start Button is Pressed

This shows the output after the start button is clicked. It shows that the current frame has no page yet because the simulation has just started. Meanwhile, on the added page it shows the page [8] which is the first page of our reference string and on the new frame it showed [8] in a blue frame because as the legend indicates it is an added frame. We can also see that it says that it was a miss and with that the page fault counter increased to 1.

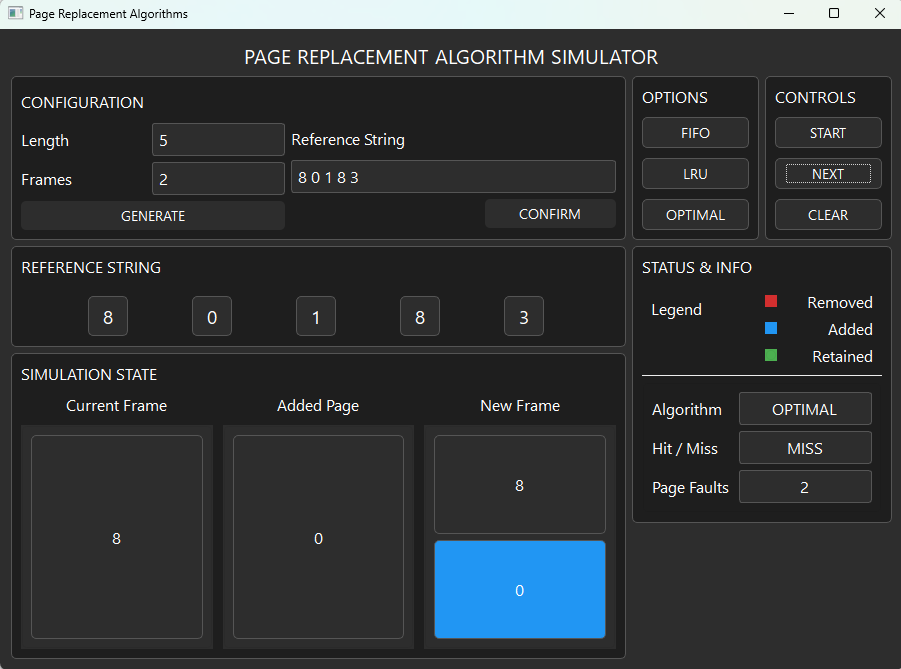


Figure 11.: The second step on the Optimal Simulation 3

In here, the second page reference which is [0] is added. The simulation state section shows the current frame has [8] and the added page is [0] which results to the new frame having [8][0] with [0] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 2.

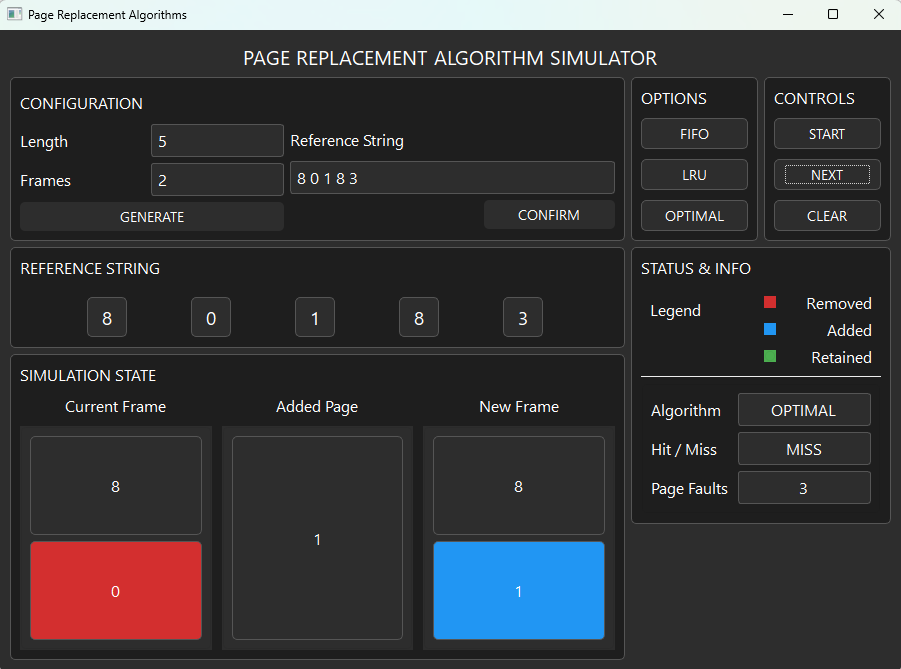


Figure 11.: The third step on the Optimal Simulation 3

In here, the third page reference which is [1] is added. The simulation state section shows the current frame has [8][0] with [0] in a red frame because it is being removed and the added page is [1] which results to the new frame having [8][1] with [1] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 3.

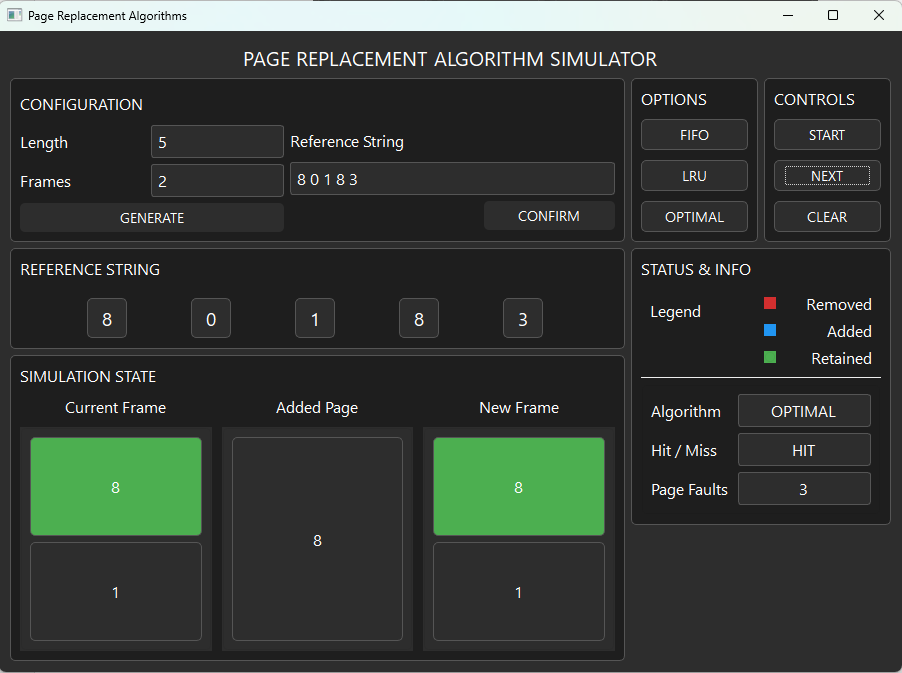


Figure 11.: The fourth step on the Optimal Simulation 3

In here, the fourth page reference which is [8] is being added which is shown on the added page section. It also shows the current frame and the new frame having [8][1] with [8] in a green frame because the added page is also a [8] which is a hit which results in the current and new frame having no changes. The Status & info section also tells us that it has a hit which makes the page fault count stay at 3.

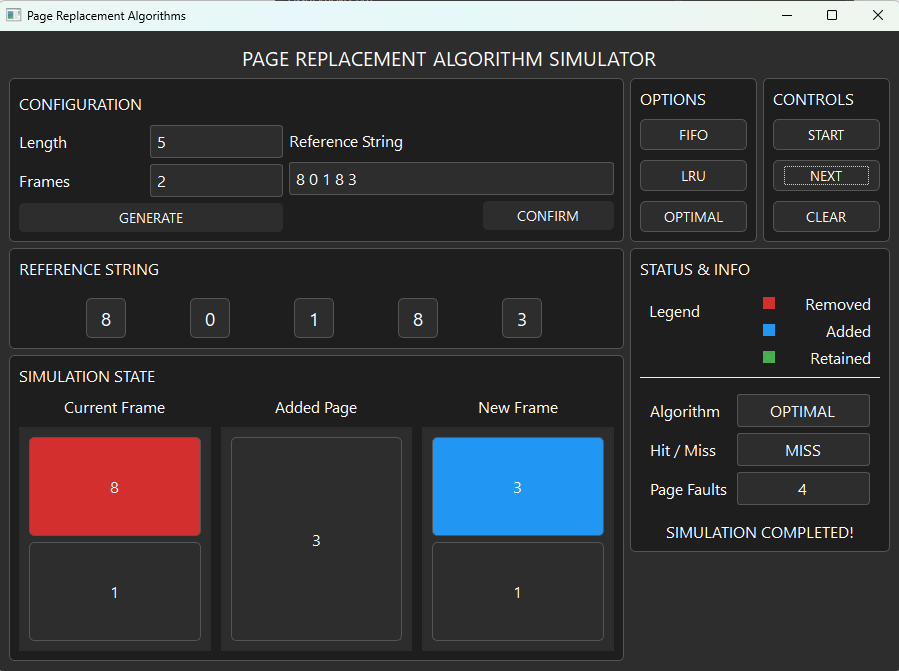


Figure 11.: The fifth step on the Optimal Simulation 3

In here, the fifth page reference which is [3] is added. The simulation state section shows the current frame has [8][1] with [8] in a red frame because it is being removed and the added page is [3] which results to the new frame having [3][1] with [3] in a blue frame because it is added. We can also see that it says that it was a miss and with that the page fault counter increased to 4. It also shows that the Simulation Completed text has been visible which states that our simulation has finished and the final number of page faults in this simulation is 4.

# Conclusion

Page replacement algorithms play a vital role in memory management within operating systems by determining which memory pages to replace when a page fault occurs. Through the simulations presented in this document, the behavior of three core algorithms—FIFO (First-In First-Out), LRU (Least Recently Used), and Optimal—was examined and compared under identical conditions.

FIFO operates in a simple, orderly fashion by removing the oldest page in memory, but it often leads to suboptimal performance due to its lack of awareness of actual page usage. LRU improves upon this by keeping track of recent usage patterns and replacing the page that has not been used for the longest time, resulting in more efficient memory utilization. Meanwhile, the Optimal algorithm, though impractical for real-world use due to its need for future knowledge, sets the benchmark for the fewest possible page faults.

From the simulations, it is evident that each algorithm has strengths and weaknesses depending on the reference string and the number of available frames. Understanding these differences helps in selecting the most appropriate algorithm for a given context and highlights the trade-offs involved in designing memory management systems. These insights are essential for students and professionals seeking a deeper understanding of operating system performance and efficiency.