Course Name: Operating systems

LAB: 12

Submitted By: Ebaad Khan

Roll: DT-22045

PROGRAM:

FIFO:

```
#include <stdio.h>


int main() {

   int i, j, k, f, pf = 0, count = 0, rs[25], m[10], n;


   printf("\nEnter the length of reference string: ");

   scanf("%d", &n);


   printf("Enter the reference string: ");

   for (i = 0; i < n; i++)

       scanf("%d", &rs[i]);


   printf("Enter number of frames: ");

   scanf("%d", &f);


   for (i = 0; i < f; i++)

       m[i] = -1;
```

```
    printf("\nPage Replacement Process:\n");
    }


    for (i = 0; i < n; i++) {
        for (k = 0; k < f; k++) {
            if (m[k] == rs[i])
                break;
        }

        if (k == f) {
            m[count] = rs[i];
            count = (count + 1) % f;
            pf++;
        }

        for (j = 0; j < f; j++)
            printf("%d\t", m[j]);
        if (k == f)
            printf("PF No. %d", pf);
        printf("\n");
    }

    printf("\nTotal Page Faults using FIFO = %d\n", pf);
    return 0;
}
```

LRU:

```
#include <stdio.h>

int main() {
    int i, j, k, min, rs[25], m[10], count[10], flag[25], n, f, pf = 0, next = 1;

    printf("Enter the length of reference string: ");
    scanf("%d", &n);

    printf("Enter the reference string: ");
    for (i = 0; i < n; i++) {
        scanf("%d", &rs[i]);
        flag[i] = 0;
    }

    printf("Enter number of frames: ");
    scanf("%d", &f);

    for (i = 0; i < f; i++) {
        m[i] = -1;
        count[i] = 0;
    }

    printf("\nPage Replacement Process:\n");

    for (i = 0; i < n; i++) {
```

```
for (j = 0; j < f; j++) {
  if (m[j] == rs[i]) {
    flag[i] = 1;
    count[j] = next++;
    break;
  }
}

if (flag[i] == 0) {
  if (i < f) {
    m[i] = rs[i];
    count[i] = next++;
  } else {
    min = 0;
    for (j = 1; j < f; j++) {
      if (count[j] < count[min])
        min = j;
    }
    m[min] = rs[i];
    count[min] = next++;
  }
  pf++;
}

for (j = 0; j < f; j++)
  printf("%d\t", m[j]);
```

```
        if (flag[i] == 0)
            printf("PF No. %d", pf);
        printf("\n");
    }

    printf("\nTotal Page Faults using LRU = %d\n", pf);
    return 0;
}
```

OPTIMAL:

```
#include <stdio.h>

int main() {
    int no_of_frames, no_of_pages, frames[10], pages[30], temp[10];
    int i, j, k, pos, max, faults = 0, flag1, flag2, flag3;

    printf("Enter number of frames: ");
    scanf("%d", &no_of_frames);

    printf("Enter number of pages: ");
    scanf("%d", &no_of_pages);

    printf("Enter page reference string: ");
    for (i = 0; i < no_of_pages; ++i)
        scanf("%d", &pages[i]);
```

```
for (i = 0; i < no_of_frames; ++i)
    frames[i] = -1;

for (i = 0; i < no_of_pages; ++i) {
    flag1 = flag2 = 0;

    for (j = 0; j < no_of_frames; ++j) {
        if (frames[j] == pages[i]) {
            flag1 = flag2 = 1;
            break;
        }
    }

    if (flag1 == 0) {
        for (j = 0; j < no_of_frames; ++j) {
            if (frames[j] == -1) {
                frames[j] = pages[i];
                faults++;
                flag2 = 1;
                break;
            }
        }
    }

    if (flag2 == 0) {
        flag3 = 0;
```

```
for (j = 0; j < no_of_frames; ++j) {

    temp[j] = -1;

    for (k = i + 1; k < no_of_pages; ++k) {

        if (frames[j] == pages[k]) {

            temp[j] = k;

            break;

        }

    }

}


for (j = 0; j < no_of_frames; ++j) {

    if (temp[j] == -1) {

        pos = j;

        flag3 = 1;

        break;

    }

}


if (flag3 == 0) {

    max = temp[0];

    pos = 0;

    for (j = 1; j < no_of_frames; ++j) {

        if (temp[j] > max) {

            max = temp[j];

            pos = j;
```

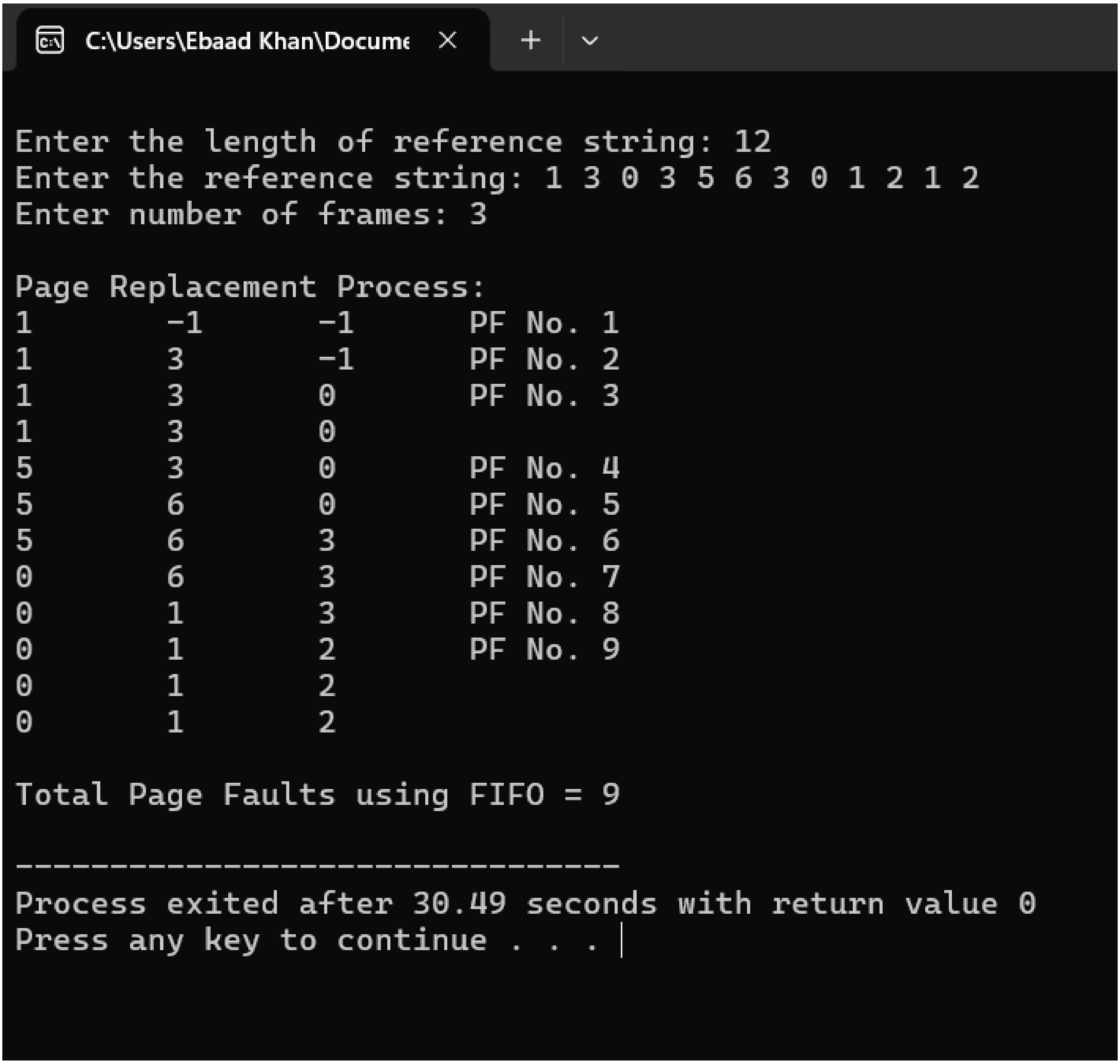
```
                    }
                }
            }

            frames[pos] = pages[i];
            faults++;
        }

        for (j = 0; j < no_of_frames; ++j)
            printf("%d\t", frames[j]);
        printf("\n");
    }

    printf("\nTotal Page Faults = %d\n", faults);
    return 0;
}
```

MRU:

```
#include <iostream>
using namespace std;

// MRU behavior: move accessed element to front
void recently(int* arr, int size, int elem) {
    int index = -1;

    for (int i = 0; i < size; i++) {
        if (arr[i] == elem) {
            index = i;
            break;
        }
    }

    if (index == -1) return; // element not found

    int temp = arr[index];
    for (int i = index; i > 0; i--)
        arr[i] = arr[i - 1];
    arr[0] = temp;
}

void print(int* arr, int size) {
    for (int i = 0; i < size; i++)
        cout << arr[i] << " ";
    cout << endl;
}

int main() {
    int arr[] = { 6, 1, 9, 5, 3 };
    int size = sizeof(arr) / sizeof(arr[0]);
    int elem = 3;

    recently(arr, size, elem);
    cout << "Array in Most Recently Used fashion: ";
    print(arr, size);


    return 0;
}
```

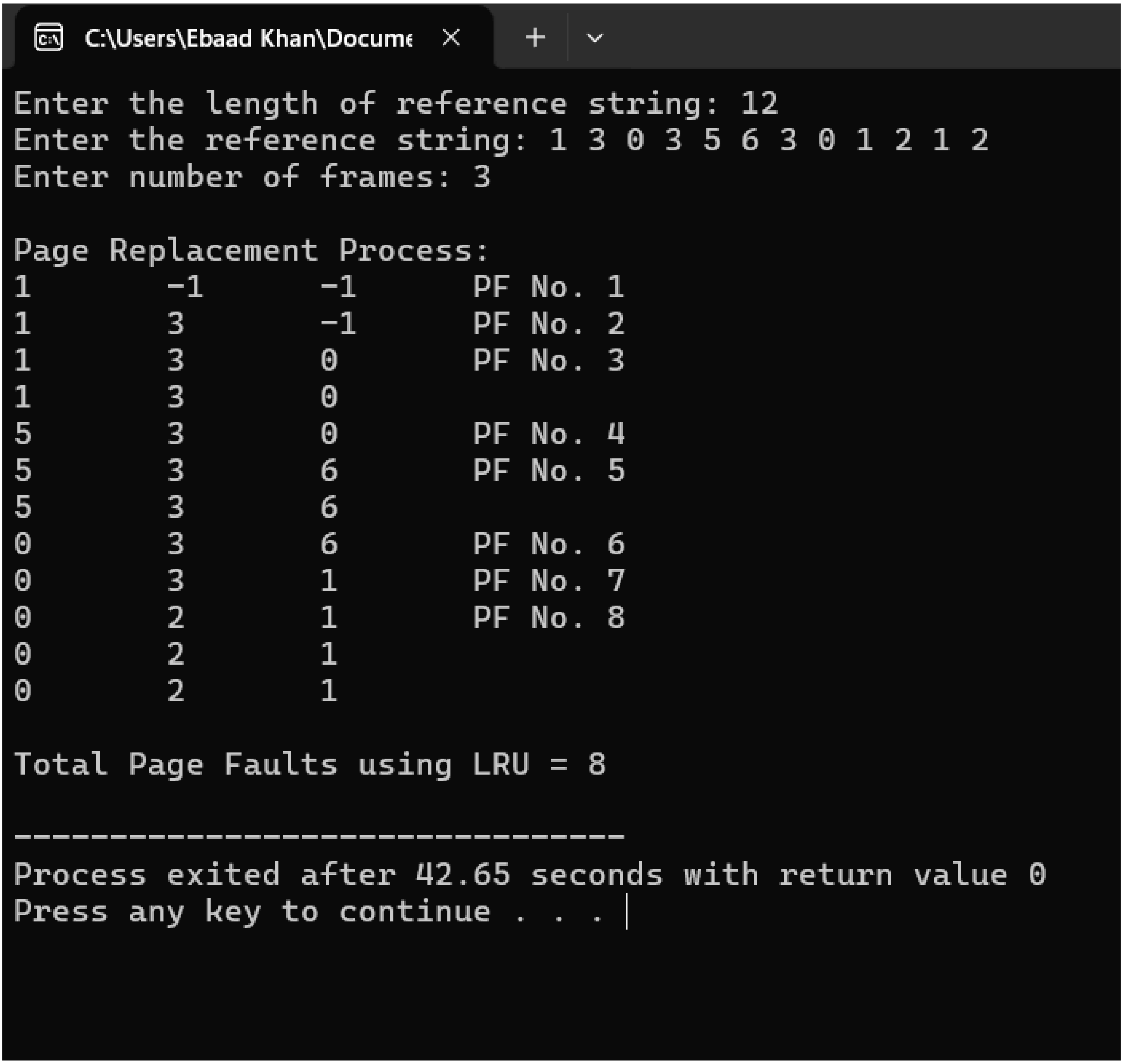
Output:

FIFO:



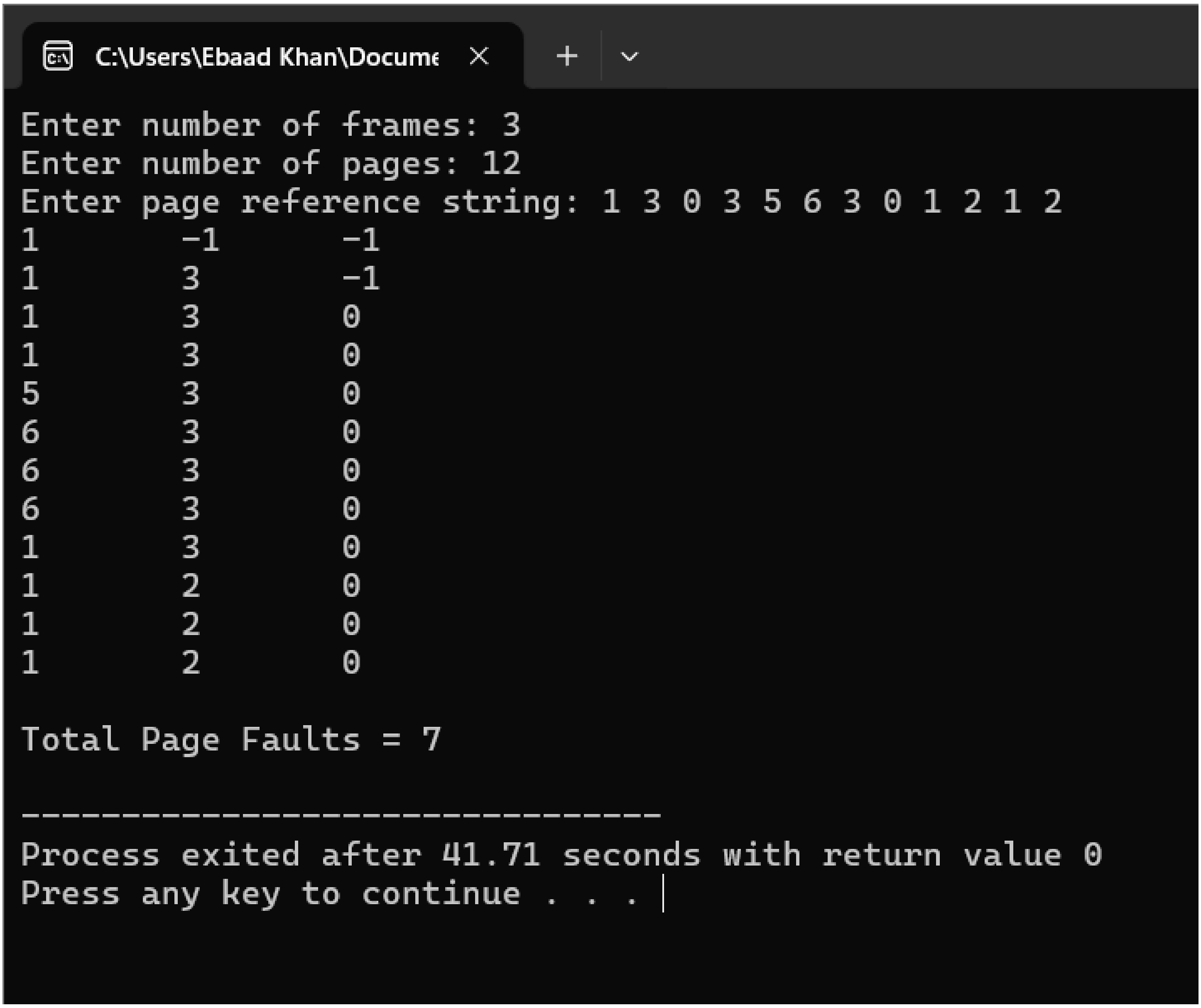
```
Enter the length of reference string: 12
Enter the reference string: 1 3 0 3 5 6 3 0 1 2 1 2
Enter number of frames: 3

Page Replacement Process:
1       -1      -1      PF No. 1
1       3       -1      PF No. 2
1       3       0       PF No. 3
1       3       0
5       3       0       PF No. 4
5       6       0       PF No. 5
5       6       3       PF No. 6
0       6       3       PF No. 7
0       1       3       PF No. 8
0       1       2       PF No. 9
0       1       2
0       1       2

Total Page Faults using FIFO = 9

--------------------------------
Process exited after 30.49 seconds with return value 0
Press any key to continue . . .
```

LRU:



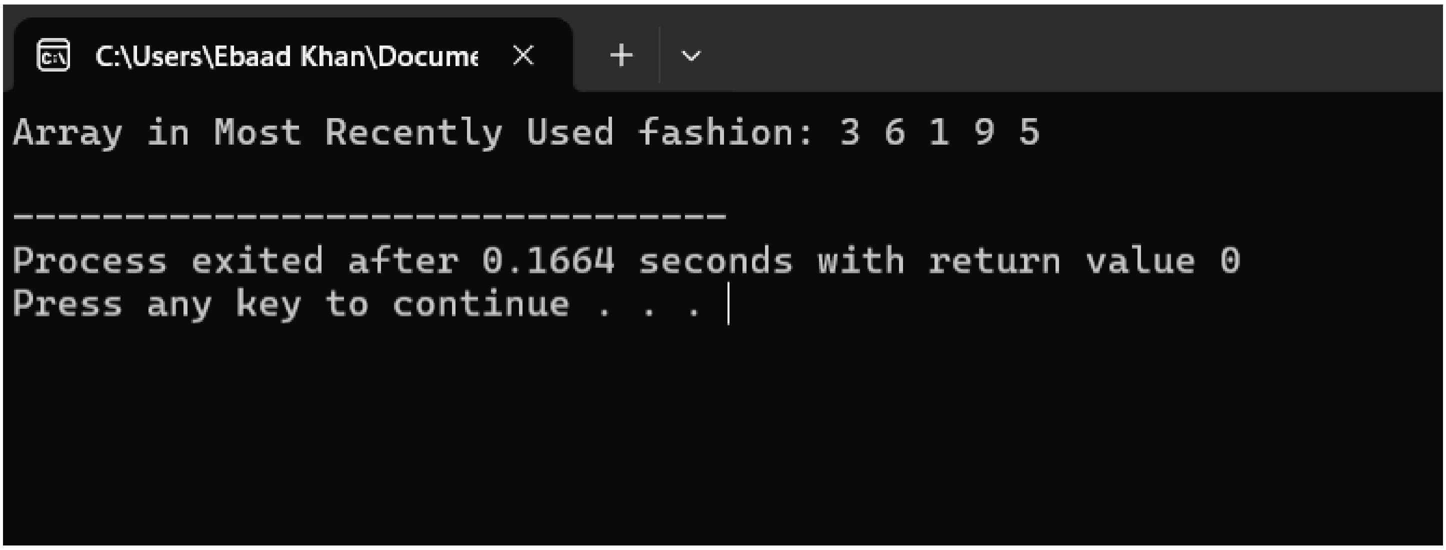
```
Enter the length of reference string: 12
Enter the reference string: 1 3 0 3 5 6 3 0 1 2 1 2
Enter number of frames: 3

Page Replacement Process:
1       -1      -1      PF No. 1
1       3       -1      PF No. 2
1       3       0       PF No. 3
1       3       0
5       3       0       PF No. 4
5       3       6       PF No. 5
5       3       6
0       3       6       PF No. 6
0       3       1       PF No. 7
0       2       1       PF No. 8
0       2       1
0       2       1

Total Page Faults using LRU = 8

--------------------------------
Process exited after 42.65 seconds with return value 0
Press any key to continue . . .
```

OPTIMAL:



```
Enter number of frames: 3
Enter number of pages: 12
Enter page reference string: 1 3 0 3 5 6 3 0 1 2 1 2
1       -1      -1
1       3       -1
1       3       0
1       3       0
5       3       0
6       3       0
6       3       0
6       3       0
1       3       0
1       2       0
1       2       0
1       2       0

Total Page Faults = 7

--------------------------------
Process exited after 41.71 seconds with return value 0
Press any key to continue . . .
```

MRU:



```
Array in Most Recently Used fashion: 3 6 1 9 5

--------------------------------
Process exited after 0.1664 seconds with return value 0
Press any key to continue . . .
```