LAB 12

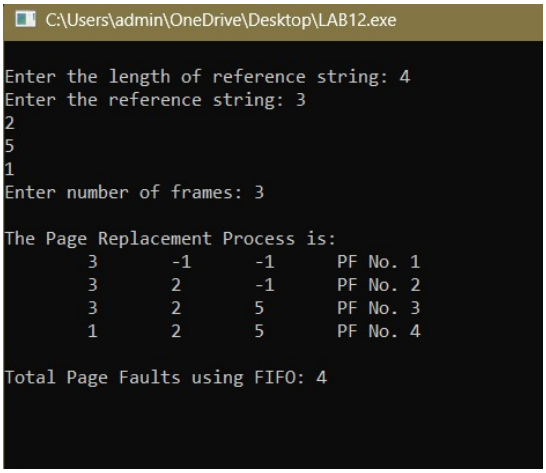
CODE:

A) FIFO

```
#include <stdio.h>
#include <conio.h>

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

   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("\nThe Page Replacement Process is:\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];
            pf++;
        }

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

        if (k == f)
            printf("\tPF No. %d", pf);
        printf("\n");

        if (count == f) count = 0;
    }

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

OUTPUT:



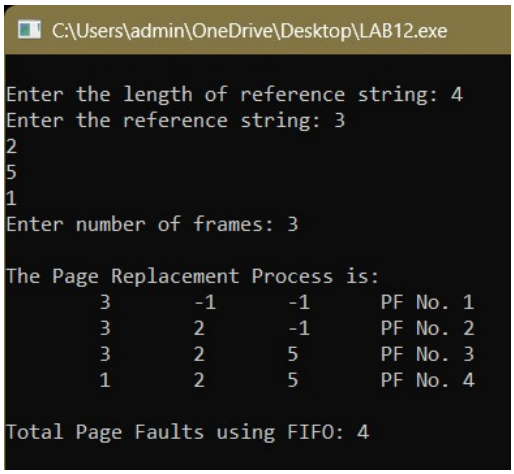
B) LRU

```
#include
<stdio.h>
#include
<conio.h>
int main() {
   int i, j, k, min, rs[25], m[10], count[10], flag[25], n, f, pf = 0, next = 1;
   clrscr();

   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++) {
      count[i] = 0;
      m[i] = -1;
   }

   printf("\nThe Page Replacement Process is:\n");
   for (i = 0; i < n; i++) {
      for (j = 0; j < f; j++) {
```

```
        if (m[j] == rs[i]) {
            flag[i] = 1;
            count[j] = next++;
        }
    }

    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[min] > count[j])
                    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);

    getch();
    return 0;
}
```

OUTPUT:



C) OPTIMAL Page Replacement

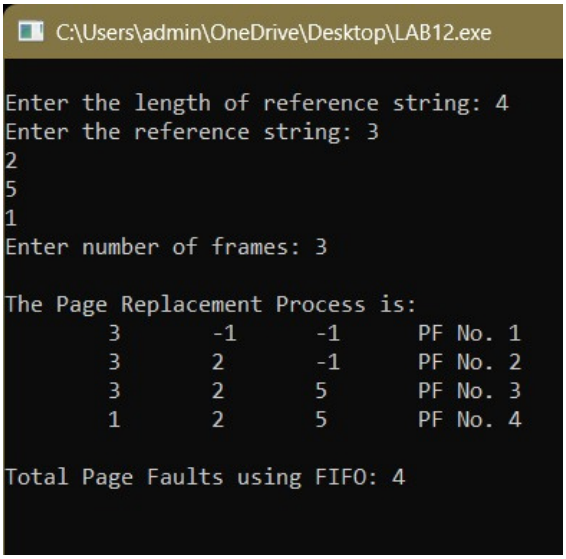
```
#include <stdio.h>

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

    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 using OPTIMAL: %d\n", faults);
    return 0;
}
```

OUTPUT:

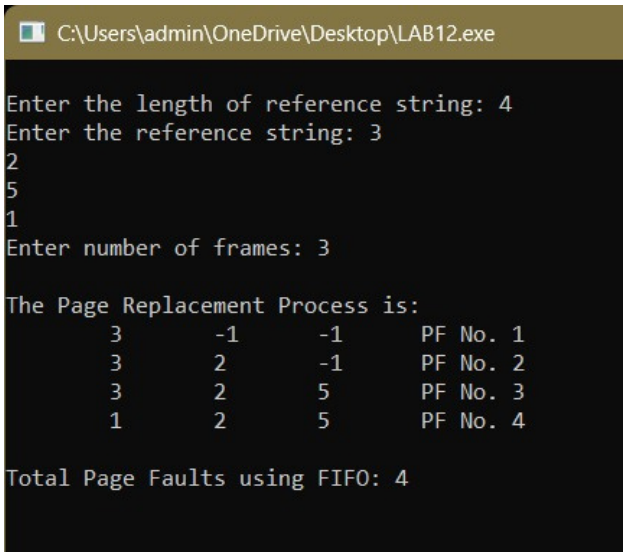


D) MRU

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

// Update array in MRU fashion
```

```
void recently(int* arr, int size, int elem)
{
    int index = elem % size;
    int temp = index, id = arr[index];

    while (temp > 0)
        arr[temp] = arr[--temp];

    arr[0] = id;
}

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

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

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

    return 0;
}
```

OUTPUT:


C:\Users\admin\OneDrive\Desktop\LAB12.exe
Enter the length of reference string: 4
Enter the reference string: 3
2
5
1
Enter number of frames: 3
The Page Replacement Process is:
3 -1 -1 PF No. 1
3 2 -1 PF No. 2
3 2 5 PF No. 3
1 2 5 PF No. 4
Total Page Faults using FIFO: 4