NAME : Arman Makhani and Raj Kariya
IDs: 202103006 and 202102048
OS LAB-9

i)FIFO

The FIFO algorithm replaces the oldest (First) page which has been present for the longest time in the main memory. In simple words, When a new page comes in from secondary memory to main memory, it selects the front of the queue which is the oldest page present and removes it.

CODE:-

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    printf("Enter the string reference length:- ");
    scanf("%d",&n);
    int str[n],i,k;
    printf("Reference String is ");
    for (i = 0; i < n; i++)
    {
        str[i] = rand() % 10;
        printf("%d ",str[i]);
    }
    int nf = 0;
    printf("\n Enter the number of frames:- ");
    scanf("%d",&nf);
    int frame[nf];
    i = 0;
    for(i = 0; i < nf; i++)
    {
        frame[i] = -1;
    }
    printf("\n The Page Replacement Process is :- \n");
    int pageFault = 0,count = 0;
    k = 0;
    i = 0;
    for ( i = 0; i < n; i++)
    {
```

```
        for ( k = 0; k < nf; k++)
        {
            if(frame[k] == str[i])
                break;
        }

        if( k == nf)
        {
            frame[count++] = str[i];
            pageFault++;
        }
        int j = 0;
        for(j =0;j<nf;j++)
            printf("\t%d",frame[j]);
        if( k == nf)
            printf("\t PF No. %d\n",pageFault);
        if(count == nf)
            count = 0;
    }
    printf("\n The number of Page Faults using FIFO are %d",pageFault);
    printf("\n The number of HITS using FIFO are %d",n-pageFault);
    printf("\n Average fault value : %f\n",((float) pageFault/n)*100);
}
```

OUTPUT:

The first output is the output that we got after running the code and the second output is from the simulator using the same reference string.

Number of frames : 4

```
Enter the string reference length:- 10
Reference String is 1 7 4 0 9 4 8 8 2 4
 Enter the number of frames:- 4

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

 The number of Page Faults using FIFO are 8
 The number of HITS using FIFO are 2
 Average fault value : 80.000000
PS C:\Users\Arman\Documents\Codes_forces>
```

First in First out (FIFO)

Frames of physical...
4

Reference String
1 7 4 0 9 4 8 8 2 4

Display options

Column li...
25

Summary

PRESETS SUBMIT

Summary - FIFO algorithm

WANT AN APP?

- Total frames: 4
- Algorithm: FIFO
- Reference string length: 10 references
- String: 1 7 4 0 9 4 8 8 2 4

Solution visualization

t	0	1	2	3	4	5	6	7	8	9	10
ref		1	7	4	0	9	4	8	8	2	4
f		1	7	4	0	9	9	8	8	2	4
f			1	7	4	0	0	9	9	8	2
f				1	7	4	4	0	0	9	8
f					1	7	7	4	4	0	9
hit		X	X	X	X	X	✓	X	✓	X	X
v						1		7		4	

- Total references: 10
- Total distinct references: 7
- Hits: 2
- Faults: 8
- **Hit rate:** 2/10 = **20%**
- **Fault rate:** 8/10 = **80%**

Number of frames : 8

```
Enter the string reference length:- 10
Reference String is 1 7 4 0 9 4 8 8 2 4
 Enter the number of frames:- 8

 The Page Replacement Process is :-
       1       -1      -1      -1      -1      -1      -1      -1       PF No. 1
       1       7       -1      -1      -1      -1      -1      -1       PF No. 2
       1       7       4       -1      -1      -1      -1      -1       PF No. 3
       1       7       4       0       -1      -1      -1      -1       PF No. 4
       1       7       4       0       9       -1      -1      -1       PF No. 5
       1       7       4       0       9       -1      -1      -1      1       7       4       0       9       8       --
1       PF No. 6
       1       7       4       0       9       8       -1      -1      1       7       4       0       9       8       2-
1       PF No. 7
       1       7       4       0       9       8       2       -1
 The number of Page Faults using FIFO are 7
 The number of HITS using FIFO are 3
 Average fault value : 70.000000
PS C:\Users\Arman\Documents\Codes_forces>
```

Algorithm

First in First out (FIFO)

Frames of physical...

8

Reference String

1 7 4 0 9 4 8 8 2 4

Display options

Column li...

25

Summary

Summary - FIFO algorithm

- Total frames: 8
- Algorithm: FIFO
- Reference string length: 10 references
- String: 1 7 4 0 9 4 8 8 2 4

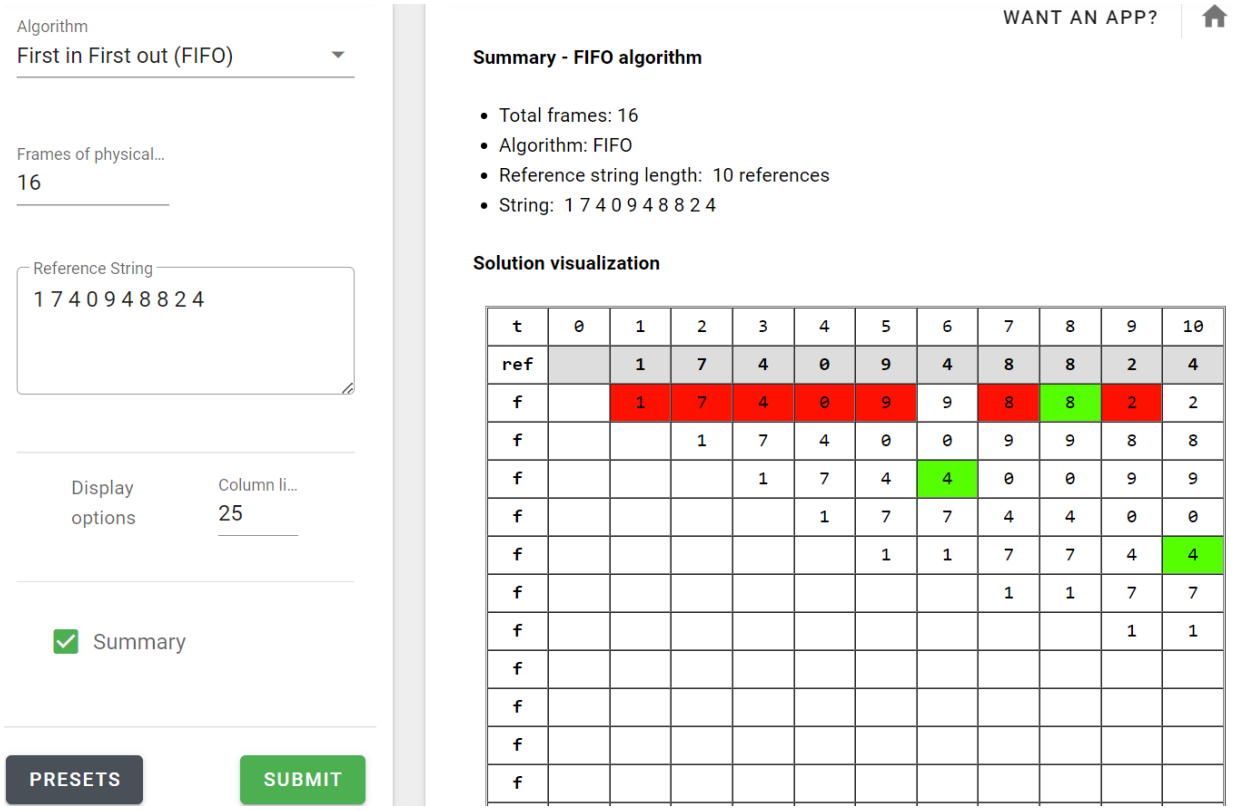
Solution visualization

t	0	1	2	3	4	5	6	7	8	9	10
ref		1	7	4	0	9	4	8	8	2	4
f		1	7	4	0	9	9	8	8	2	2
f			1	7	4	0	0	9	9	8	8
f				1	7	4	4	0	0	9	9
f					1	7	7	4	4	0	0
f						1	1	7	7	4	4
f								1	1	7	7
f										1	1
f											
hit		✗	✗	✗	✗	✗	✓	✗	✓	✗	✓
v											

- Total references: 10
- Total distinct references: 7
- Hits: 3
- Faults: 7
- **Hit rate:** 3/10 = **30**%
- **Fault rate:** 7/10 = **70**%

Number of frames : 16

```
Enter the string reference length:- 10
Reference String is 1 7 4 0 9 4 8 8 2 4
 Enter the number of frames:- 16

 The Page Replacement Process is :-
       1       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      --
1       PF No. 1
       1       7       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      --
1       PF No. 2
       1       7       4       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      --
1       PF No. 3
       1       7       4       0       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      --
1       PF No. 4
       1       7       4       0       9       -1      -1      -1      -1      -1      -1      -1      -1      -1      --
1       PF No. 5
       1       7       4       0       9       -1      -1      -1      -1      -1      -1      -1      -1      -1      --
1      1       7       4       0       9       8       -1      -1      -1      -1      -1      -1      -1      -1      --
1       PF No. 6
       1       7       4       0       9       8       -1      -1      -1      -1      -1      -1      -1      -1      --
1      1       7       4       0       9       8       2       -1      -1      -1      -1      -1      -1      -1      --
1       PF No. 7
       1       7       4       0       9       8       2       -1      -1      -1      -1      -1      -1      -1      --
1
 The number of Page Faults using FIFO are 7
 The number of HITS using FIFO are 3
 Average fault value : 70.000000
PS C:\Users\Arman\Documents\Codes_forces>
```



t	0	1	2	3	4	5	6	7	8	9	10
ref		1	7	4	0	9	4	8	8	2	4
f		1	7	4	0	9	9	8	8	2	2
f			1	7	4	0	0	9	9	8	8
f				1	7	4	4	0	0	9	9
f					1	7	7	4	4	0	0
f						1	1	7	7	4	4
f								1	1	7	7
f										1	1
f											
f											
f											
f											

- Total references: 10
- Total distinct references: 7
- Hits: 3
- Faults: 7
- **Hit rate:** 3/10 = **30%**
- **Fault rate:** 7/10 = **70%**

II)LRU:-

In this algorithm page will be replaced which is least recently used. It uses the page which has not been referenced for a long time in the past and replaces it.

CODE:-

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
```

```
    int n;
    printf("Enter the string reference length:- ");
    scanf("%d",&n);
    int str[n],i,k,flag[n];
    printf("Reference String is ");
    for (i = 0; i < n; i++)
    {
        str[i] = rand() % 10;
        printf("%d ",str[i]);
        flag[i] = 0;
    }
    int nf = 0;
    printf("\n Enter the number of frames:- ");
    scanf("%d",&nf);
    int frame[nf],count[10];
    i = 0;
    for(i = 0; i < nf; i++)
    {
        frame[i] = -1;
        count[i] = 0;
    }
    printf("\n The Page Replacement Process is :- \n");
    int pageFault = 0,next = 0;
    k = 0;
    i = 0;
    int j = 0,mini;
    for( i = 0; i < n; i++)
    {

        for ( k = 0; k < nf; k++)
        {
            if(frame[k] == str[i])
            {
                flag[i] = 1;
                count[k] = next;
                next++;
            }
        }
        if(flag[i] == 0)
        {
```

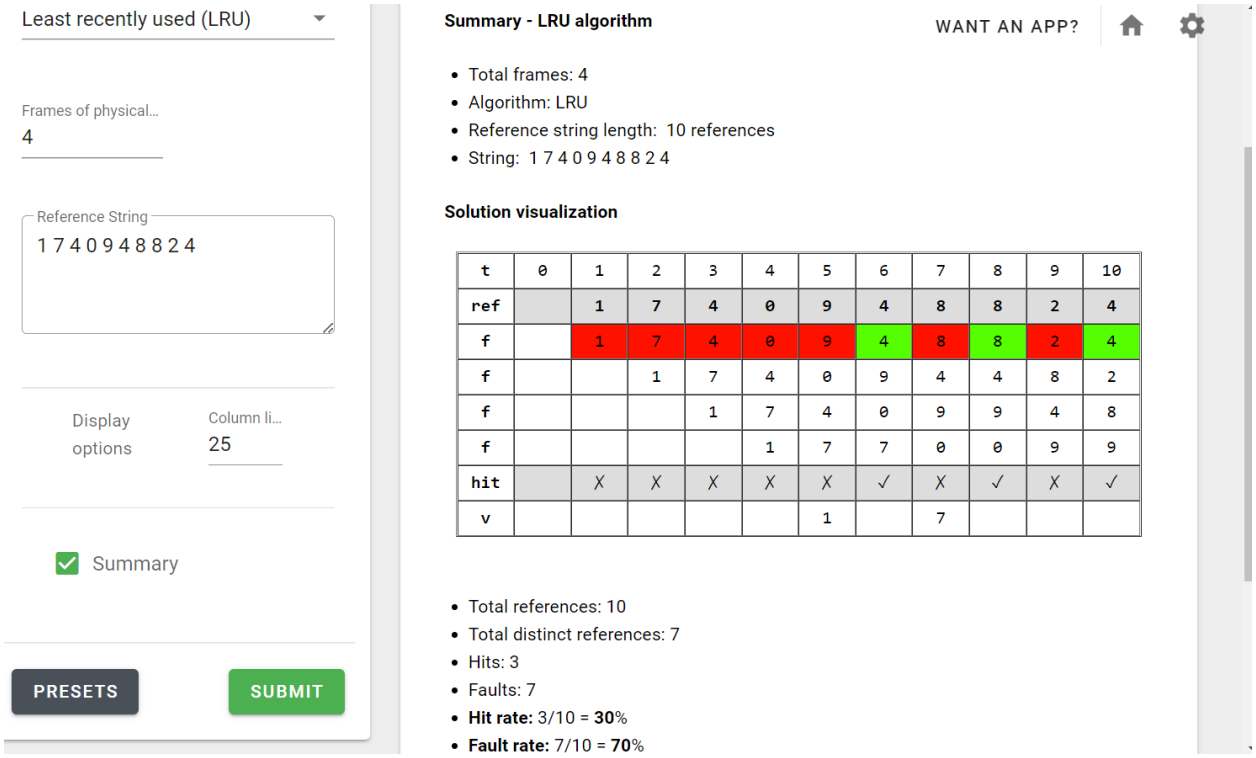
```
            if( i < nf)
            {
                frame[i] = str[i];
                count[i] = next;
                next++;
            }
            else
            {
                mini = 0;
                for(j = 0; j < nf; j++)
                {
                    if(count[mini] > count[j])
                    {
                        mini = j;
                    }
                }
                frame[mini] = str[i];
                count[mini] = next;
                next++;
            }
            pageFault++;
        }
        printf("\n");
        for(j = 0; j < nf;j++)
        {
            printf("%d\t",frame[j]);
        }
        if(flag[i] == 0)
        {
            printf("PF No. :- %d",pageFault);
            printf("\n");
        }
    }
    printf("\n The number of Page Faults using LRU are %d",pageFault);
    printf("\n The number of HITS using LRU are %d",n-pageFault);
    printf("\n Average fault value : %f\n",((float) pageFault/n)*100);
}
```

OUTPUT:

First output is the output that we got after running the code and the second output is from the simulator using the same reference string.

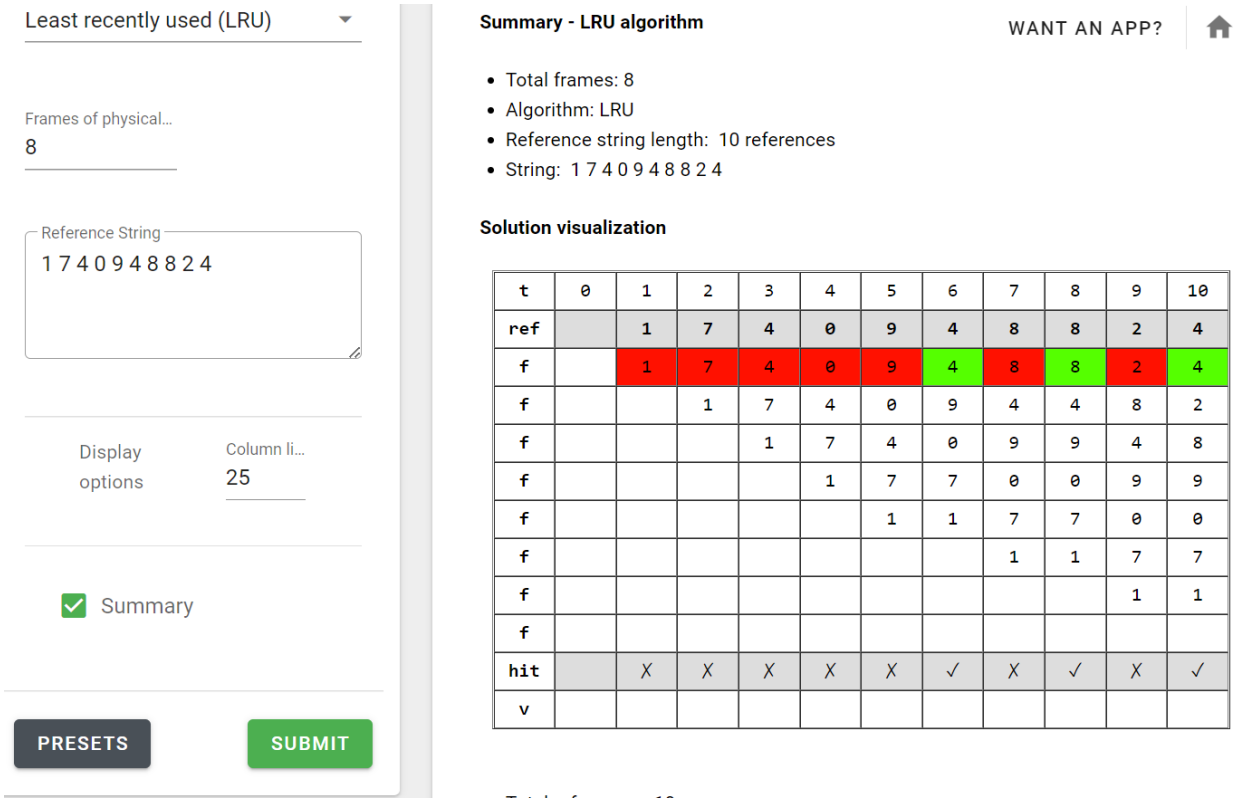
No. of frames: 4

```
Enter the string reference length:- 10
Reference String is 1 7 4 0 9 4 8 8 2 4
 Enter the number of frames:- 4

 The Page Replacement Process is :-

1       -1      -1      -1      PF No. :- 1

1       7       -1      -1      PF No. :- 2

1       7       4       -1      PF No. :- 3

1       7       4       0       PF No. :- 4

9       7       4       0       PF No. :- 5

9       7       4       0
9       8       4       0       PF No. :- 6

9       8       4       0
9       8       4       2       PF No. :- 7

9       8       4       2
 The number of Page Faults using LRU are 7
 The number of HITS using LRU are 3
 Average fault value : 70.000000
PS C:\Users\Arman\Documents\Codes_forces>
```



No. of frames : 8

```
Enter the string reference length:- 10
Reference String is 1 7 4 0 9 4 8 8 2 4
 Enter the number of frames:- 8

 The Page Replacement Process is :-

1       -1      -1      -1      -1      -1      -1      -1      PF No. :- 1

1       7       -1      -1      -1      -1      -1      -1      PF No. :- 2

1       7       4       -1      -1      -1      -1      -1      PF No. :- 3

1       7       4       0       -1      -1      -1      -1      PF No. :- 4

1       7       4       0       9       -1      -1      -1      PF No. :- 5

1       7       4       0       9       -1      -1      -1
1       7       4       0       9       -1      8       -1      PF No. :- 6

1       7       4       0       9       -1      8       -1
2       7       4       0       9       -1      8       -1      PF No. :- 7

2       7       4       0       9       -1      8       -1
 The number of Page Faults using LRU are 7
 The number of HITS using LRU are 3
 Average fault value : 70.000000
PS C:\Users\Arman\Documents\Codes_forces>
```

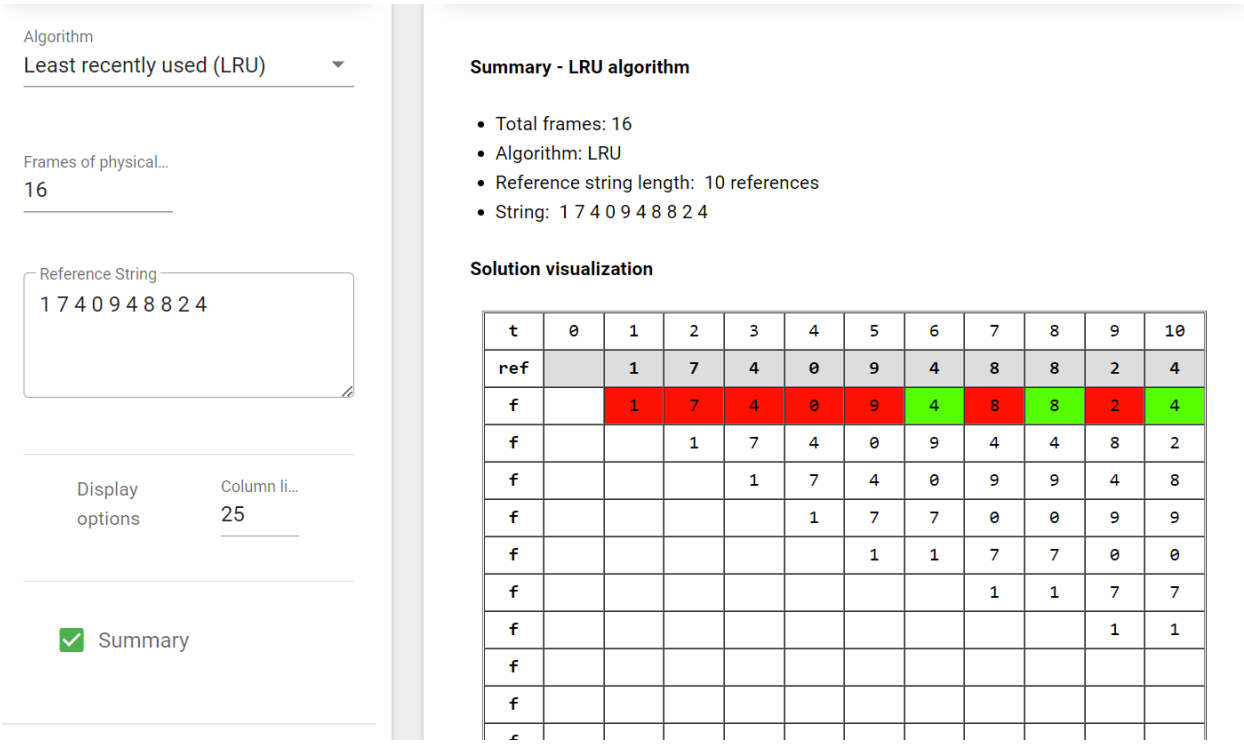


t	0	1	2	3	4	5	6	7	8	9	10
ref		1	7	4	0	9	4	8	8	2	4
f		1	7	4	0	9	4	8	8	2	4
f			1	7	4	0	9	4	4	8	2
f				1	7	4	0	9	9	4	8
f					1	7	7	0	0	9	9
f						1	1	7	7	0	0
f								1	1	7	7
f										1	1
f											
hit		✗	✗	✗	✗	✗	✓	✗	✓	✗	✓
v											

- Total references: 10
- Total distinct references: 7
- Hits: 3
- Faults: 7
- **Hit rate:** 3/10 = **30%**
- **Fault rate:** 7/10 = **70%**

No. of frames:16

```
Reference String is 1 7 4 0 9 4 8 8 2 4
 Enter the number of frames:- 16

 The Page Replacement Process is :-

PF No. :- 1

PF No. :- 2

PF No. :- 3

PF No. :- 4

PF No. :- 5

PF No. :- 6

PF No. :- 7

PF No. :- 8

PF No. :- 9

PF No. :- 10

 The number of Page Faults using LRU are 10
 The number of HITS using LRU are 0
 Average fault value : 100.000000
PS C:\Users\Arman\Documents\Codes_forces>
```



t	0	1	2	3	4	5	6	7	8	9	10
ref		1	7	4	0	9	4	8	8	2	4
f		1	7	4	0	9	4	8	8	2	4
f			1	7	4	0	9	4	4	8	2
f				1	7	4	0	9	9	4	8
f					1	7	7	0	0	9	9
f						1	1	7	7	0	0
f								1	1	7	7
f										1	1
f											
f											
f											

- Total references: 10
- Total distinct references: 7
- Hits: 3
- Faults: 7
- **Hit rate:** 3/10 = **30%**
- **Fault rate:** 7/10 = **70%**

III) OPT:-

In OPT when a page replacement is needed, it looks ahead in the input queue for the page frame which will be referenced only after a long time. The page with the longest reference is swapped.

CODE:-

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    printf("Enter the string reference length:- ");
    scanf("%d",&n);
    int str[n],i,k;
    printf("Reference String is ");
    for (i = 0; i < n; i++)
    {
        str[i] = rand() % 10;
        printf("%d ",str[i]);
    }
    int nf = 0;
    printf("\n Enter the number of frames:- ");
    scanf("%d",&nf);
    int frame[nf],temp[10];
    i = 0;
    for(i = 0; i < nf; i++)
    {
        frame[i] = -1;
    }
    printf("\n The Page Replacement Process is :- \n");
    int Fault = 0,count = 0,flag1,flag2,flag3,pos,maxi;
    k = 0;
    i = 0;
    int j = 0;
    for( i = 0; i < n; i++)
    {
        flag1 = flag2 = 0;
        for ( k = 0; k < nf; k++)
        {
            if(frame[k] == str[i])
            {
                flag1 = flag2 = 1;
                break;
            }
        }
        if(flag1 == 0)
```

```
        {
            for(j =0;j < nf;j++)
            {
                if(frame[j] == -1)
                {
                    Fault++;
                    frame[j] = str[i];
                    flag2 = 1;
                    break;
                }
            }
        }
        if(flag2 == 0)
        {
            flag3 = 0;
            for(j = 0; j < nf;j++)
            {
                temp[j] = -1;
                for(k = i+1;k<n;k++)
                {
                    if(frame[j] == str[i])
                    {
                        temp[j] = k;
                        break;
                    }
                }
            }
            for(j = 0; j < nf;j++)
            {
                if(temp[j] == -1)
                {
                    pos = j;
                    flag3 = 1;
                    break;
                }
            }
            if(flag3 == 0)
            {
                maxi = temp[0];
                pos = 0;
```

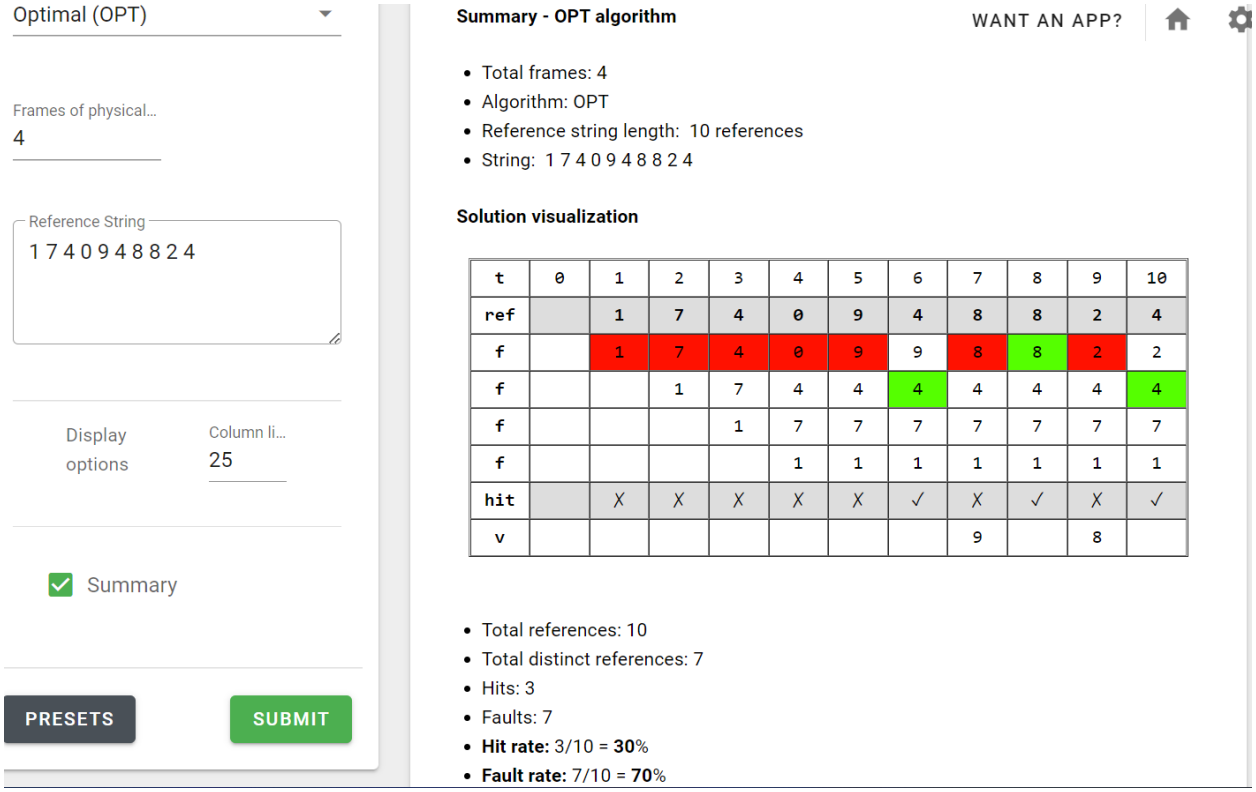
```
                for(j = 0; j < nf;j++)
                {
                    if(temp[j] > maxi)
                    {
                        maxi = temp[j];
                        pos = j;
                    }
                }
            }
            frame[pos] = str[i];
            Fault++;
        }
        printf("\n");
        for(j = 0; j < nf;j++)
        {
            printf("%d\t",frame[j]);
        }
    }
    printf("\n The number of Page Faults using OPT are %d",Fault);
    printf("\n The number of HITS using OPT are %d",n-Fault);
    printf("\n Average fault value : %f\n",((float) Fault/n)*100);
}
```

OUTPUT:

First output is the output that we got after running the code and the second output is from the simulator using the same reference string.

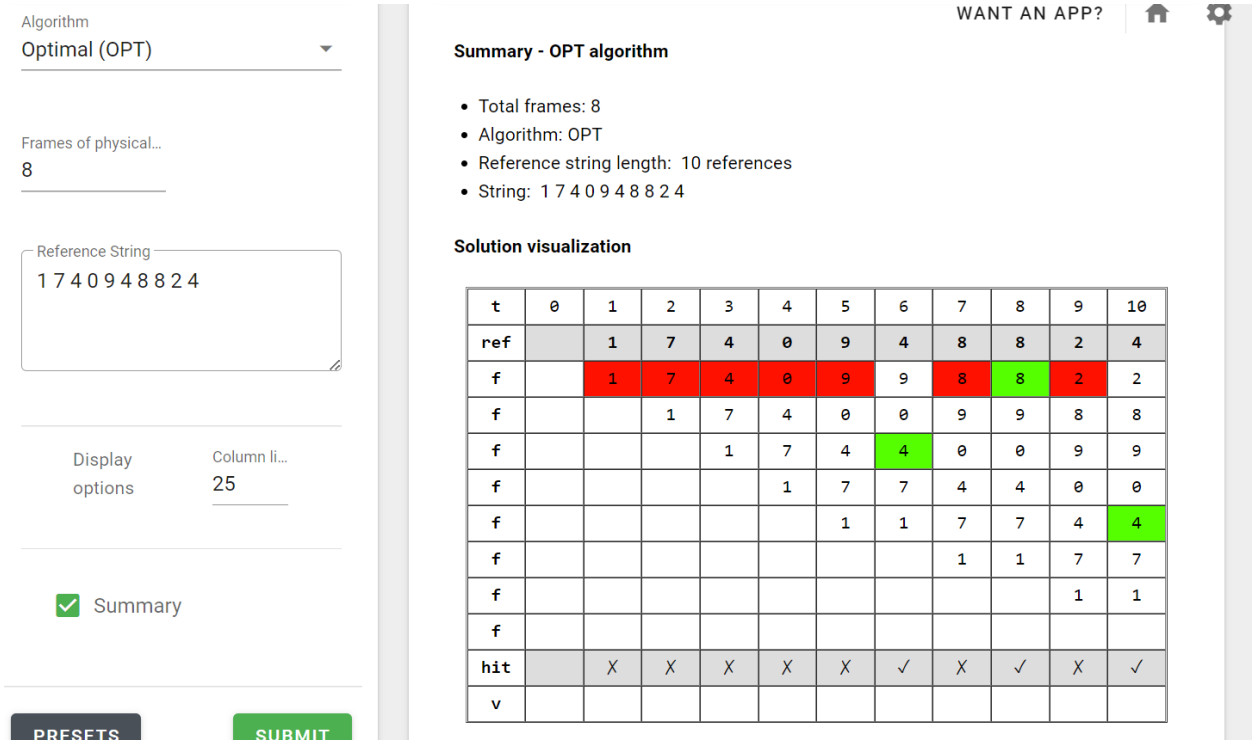
No. of frames: 4

```
Enter the string reference length:- 10
Reference String is 1 7 4 0 9 4 8 8 2 4
 Enter the number of frames:- 4

 The Page Replacement Process is :-

1       -1      -1      -1
1       7       -1      -1
1       7       4       -1
1       7       4       0
9       7       4       0
9       7       4       0
8       7       4       0
8       7       4       0
2       7       4       0
2       7       4       0
 The number of Page Faults using OPT are 7
 The number of HITS using OPT are 3
 Average fault value : 70.000000
PS C:\Users\Arman\Documents\Codes_forces>
```



No. of frames : 8

```
Enter the string reference length:- 10
Reference String is 1 7 4 0 9 4 8 8 2 4
 Enter the number of frames:- 8

 The Page Replacement Process is :-

1       -1      -1      -1      -1      -1      -1      -1
1       7       -1      -1      -1      -1      -1      -1
1       7       4       -1      -1      -1      -1      -1
1       7       4       0       -1      -1      -1      -1
1       7       4       0       9       -1      -1      -1
1       7       4       0       9       -1      -1      -1
1       7       4       0       9       8       -1      -1
1       7       4       0       9       8       -1      -1
1       7       4       0       9       8       2       -1
1       7       4       0       9       8       2       -1
 The number of Page Faults using OPT are 7
 The number of HITS using OPT are 3
 Average fault value : 70.000000
```

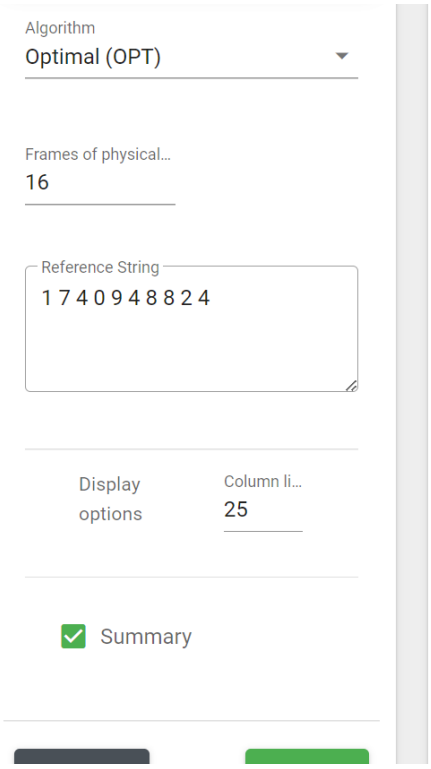


t	0	1	2	3	4	5	6	7	8	9	10
ref		1	7	4	0	9	4	8	8	2	4
f		1	7	4	0	9	9	8	8	2	2
f			1	7	4	0	0	9	9	8	8
f				1	7	4	4	0	0	9	9
f					1	7	7	4	4	0	0
f						1	1	7	7	4	4
f								1	1	7	7
f										1	1
f											
hit		✗	✗	✗	✗	✗	✓	✗	✓	✗	✓
v											

- Total references: 10
- Total distinct references: 7
- Hits: 3
- Faults: 7
- **Hit rate:** 3/10 = **30%**
- **Fault rate:** 7/10 = **70%**

No. of frames:16

```
Enter the string reference length:- 10
Reference String is 1 7 4 0 9 4 8 8 2 4
 Enter the number of frames:- 16

 The Page Replacement Process is :-

1       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1
1       7       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1
1       7       4       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1
1       7       4       0       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1
1       7       4       0       9       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1
1       7       4       0       9       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1      -1
1       7       4       0       9       8       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1
1       7       4       0       9       8       -1      -1      -1      -1      -1      -1      -1      -1      -1      -1
1       7       4       0       9       8       2       -1      -1      -1      -1      -1      -1      -1      -1      -1
1       7       4       0       9       8       2       -1      -1      -1      -1      -1      -1      -1      -1      -1
 The number of Page Faults using OPT are 7
 The number of HITS using OPT are 3
 Average fault value : 70.000000
PS C:\Users\Arman\Documents\Codes_forces>
```



Summary - OPT algorithm

- Total frames: 16
- Algorithm: OPT
- Reference string length: 10 references
- String: 1 7 4 0 9 4 8 8 2 4

Solution visualization

t	0	1	2	3	4	5	6	7	8	9	10
ref		1	7	4	0	9	4	8	8	2	4
f		1	7	4	0	9	9	8	8	2	2
f			1	7	4	0	0	9	9	8	8
f				1	7	4	4	0	0	9	9
f					1	7	7	4	4	0	0
f						1	1	7	7	4	4
f								1	1	7	7
f										1	1
f											
f											
f											

- Total references: 10
- Total distinct references: 7
- Hits: 3
- Faults: 7
- **Hit rate:** 3/10 = **30%**
- **Fault rate:** 7/10 = **70%**

CONCLUSION:

Number of fault pages are highest is FIFO , second highest in LRU and least in OPT.