PAGE REPLACEMENT ALGORITHMS

ROLL NO: 187156

1. **FIFO**

**Code:**

#include<bits/stdc++.h>

using namespace std;

void display(vector<int>v)

{

for(auto i:v)

cout<<i<<" ";

cout<<endl;

}

int solve(int A[], int n, int k)

{

int cc = 0;

vector<int>v1, v2;

int i = 0;

while(i < k)

{

for(int j=0;j<v2.size();j++)

v2[j]++;

if(find(v1.begin(), v1.end(), A[i]) == v1.end()){

cout<<"Current Element is "<<A[i]<<" And "<<"The Frame Window is ";

display(v1);

v1.push\_back(A[i]);

v2.push\_back(1);

cc++;

}

i++;

}

while(i<n)

{

for(int j=0;j<v2.size();j++)

v2[j]++;

if(find(v1.begin(), v1.end(), A[i]) == v1.end()){

cout<<"Current Element is "<<A[i]<<" And "<<"The Frame Window is ";

display(v1);

int m1 = -1, m2 = INT\_MIN;

for(int j=0;j<v2.size();j++){

if(v2[j] > m2){

m2 = v2[j];

m1 = j;

}

}

v1[m1] = A[i];

v2[m1] = 1;

cc++;

}

i++;

}

return cc;

}

int main()

{

int n;

cout<<"Enter string length: ";

cin>>n;

int A[n];

cout<<"Enter string elements: ";

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

{

cin>>A[i];

}

cout<<"Enter frame length: ";

int k;

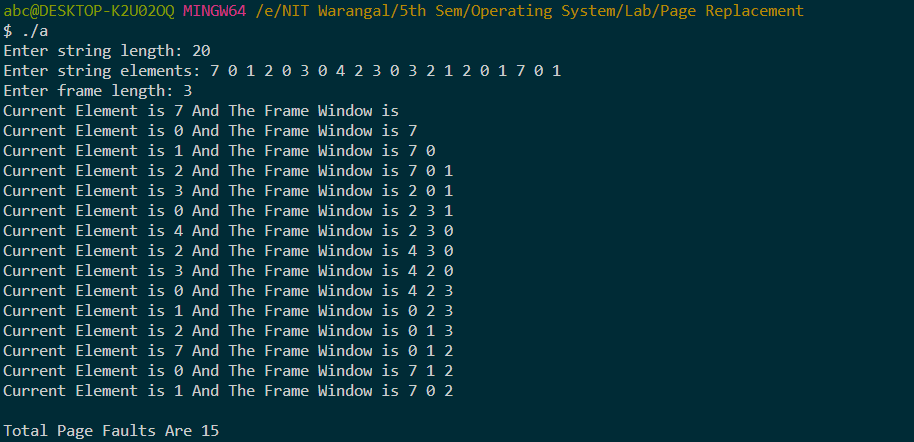
cin>>k;

cout<<"\nTotal Page Faults Are "<<solve(A, n, k);

return 0;

}

**Output:**

****

1. **Optimal**

**Code:**

#include<bits/stdc++.h>

using namespace std;

void display(vector<int>v)

{

for(auto i:v)

cout<<i<<" ";

cout<<endl;

}

int solve(int A[], int n, int k)

{

int cc = 0;

map<int, vector<int>>mp;

map<int, vector<int>>::iterator it;

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

{

mp[A[i]].push\_back(i);

}

for(it = mp.begin();it != mp.end();it++)

{

it->second.push\_back(100);

}

vector<int>v1;

int i = 0;

while(v1.size() < k)

{

if(find(v1.begin(), v1.end(), A[i]) == v1.end()){

cout<<"Current Element is "<<A[i]<<" And "<<"The Frame Window is ";

for(int j=0;j<v1.size();j++){

cout<<v1[j]<<" ";

}

cout<<endl;

v1.push\_back(A[i]);

cc++;

}

i++;

}

while(i<n){

cout<<"Current Element is "<<A[i]<<" And "<<"The Frame Window is ";

for(int j=0;j<v1.size();j++){

cout<<v1[j]<<" ";

}

cout<<endl;

if(find(v1.begin(), v1.end(), A[i]) == v1.end()){

cc++;

int m1 = -1, m2 = INT\_MIN;

for(int j=0;j<v1.size();j++){

int m3 = 0;

for(int l=0;l<mp[v1[j]].size();l++){

if(mp[v1[j]][l] > i){

m3 = mp[v1[j]][l];

break;

}

}

// cout<<v1[j]<<"->"<<m3<<", "<<m2<<endl;

if(m3 > m2){

m2 = m3;

m1 = j;

}

}

// cout<<"Max is "<<m1<<endl;

v1[m1] = A[i];

}

i++;

}

return cc;

}

int main()

{

int n;

cout<<"Enter string length: ";

cin>>n;

int A[n];

cout<<"Enter string elements: ";

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

{

cin>>A[i];

}

cout<<"Enter frame length: ";

int k;

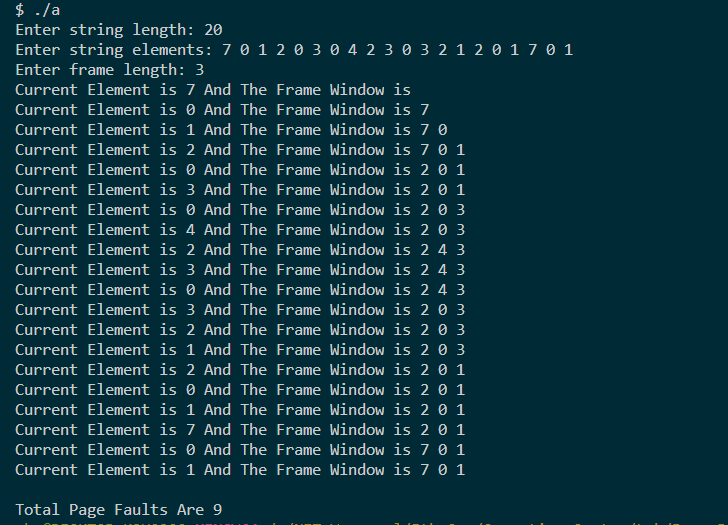
cin>>k;

cout<<"\nTotal Page Faults Are "<<solve(A, n, k);

return 0;

}

**Output:**

****

1. **Least Recently Used (LRU)**

**Code:**

#include<bits/stdc++.h>

using namespace std;

void display(list<int>l)

{

cout<<"Frame window is ";

for(auto i:l)

cout<<i<<" ";

cout<<endl;

}

int solve(int A[], int n, int k)

{

int cc = 0;

list<int>l;

map<int, list<int>::iterator>mp;

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

{

if(mp.find(A[i]) == mp.end())

{

cout<<"Element is "<<A[i]<<" And ";

display(l);

cc++;

if(l.size() == k)

{

int last = l.back();

l.pop\_back();

mp.erase(last);

}

}

else

{

l.erase(mp[A[i]]);

}

l.push\_front(A[i]);

mp[A[i]] = l.begin();

}

return cc;

}

int main()

{

int n;

cout<<"Enter string length: ";

cin>>n;

int A[n];

cout<<"Enter string elements: ";

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

{

cin>>A[i];

}

cout<<"Enter frame length: ";

int k;

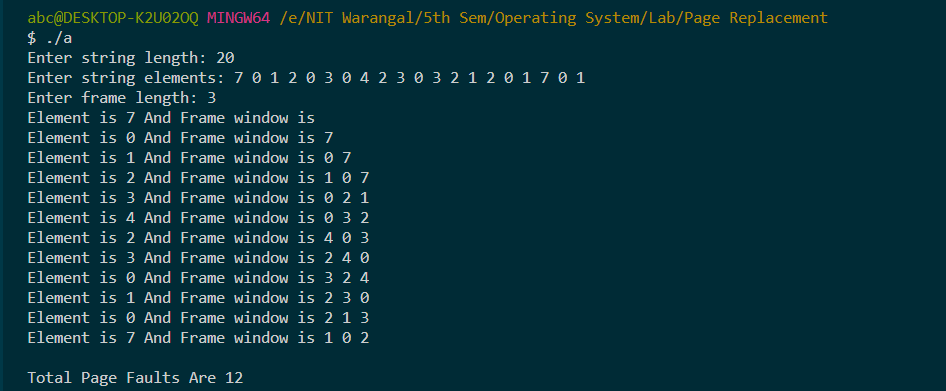
cin>>k;

cout<<"\nTotal Page Faults Are "<<solve(A, n, k);

return 0;

}

**Output:**

****