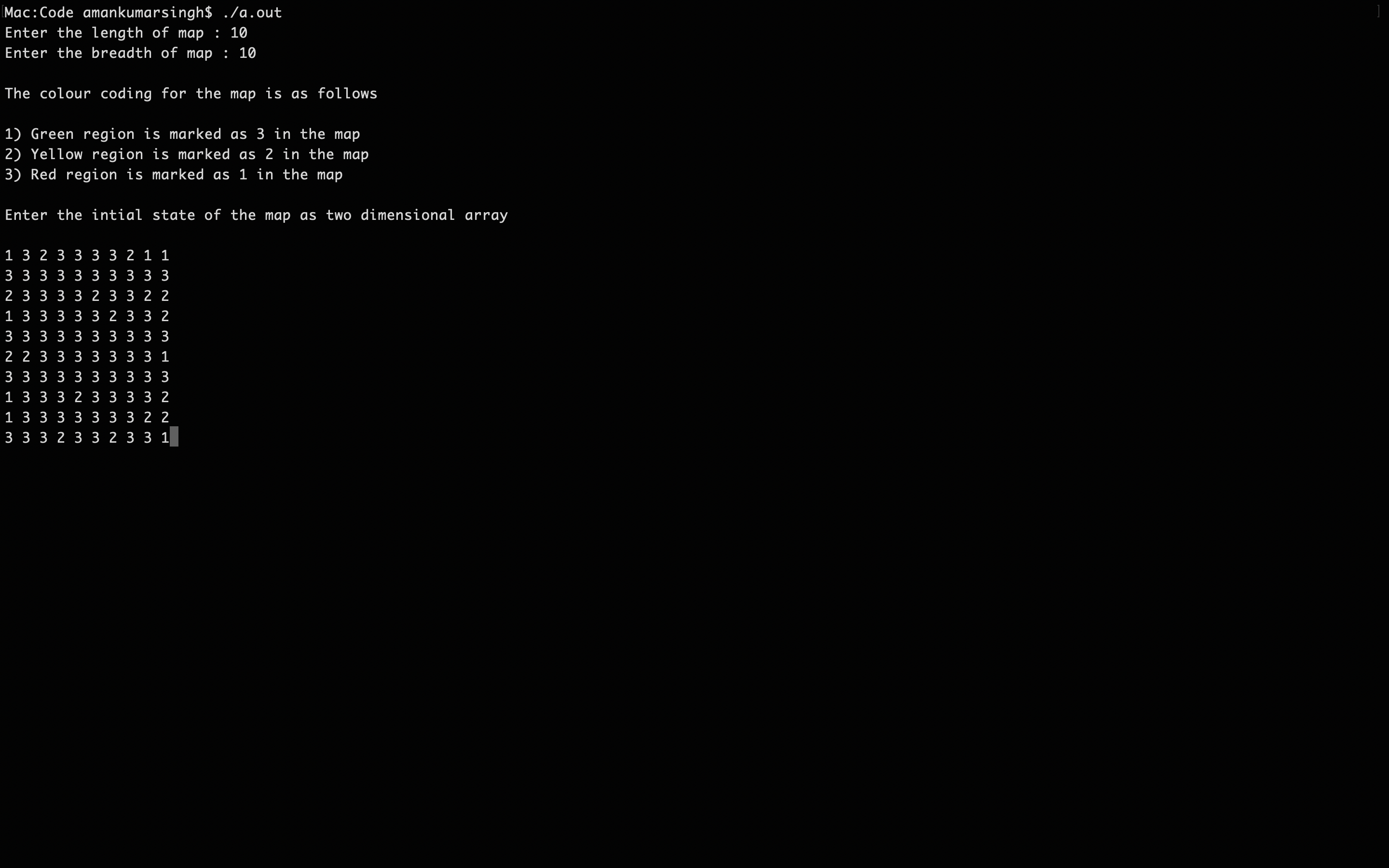
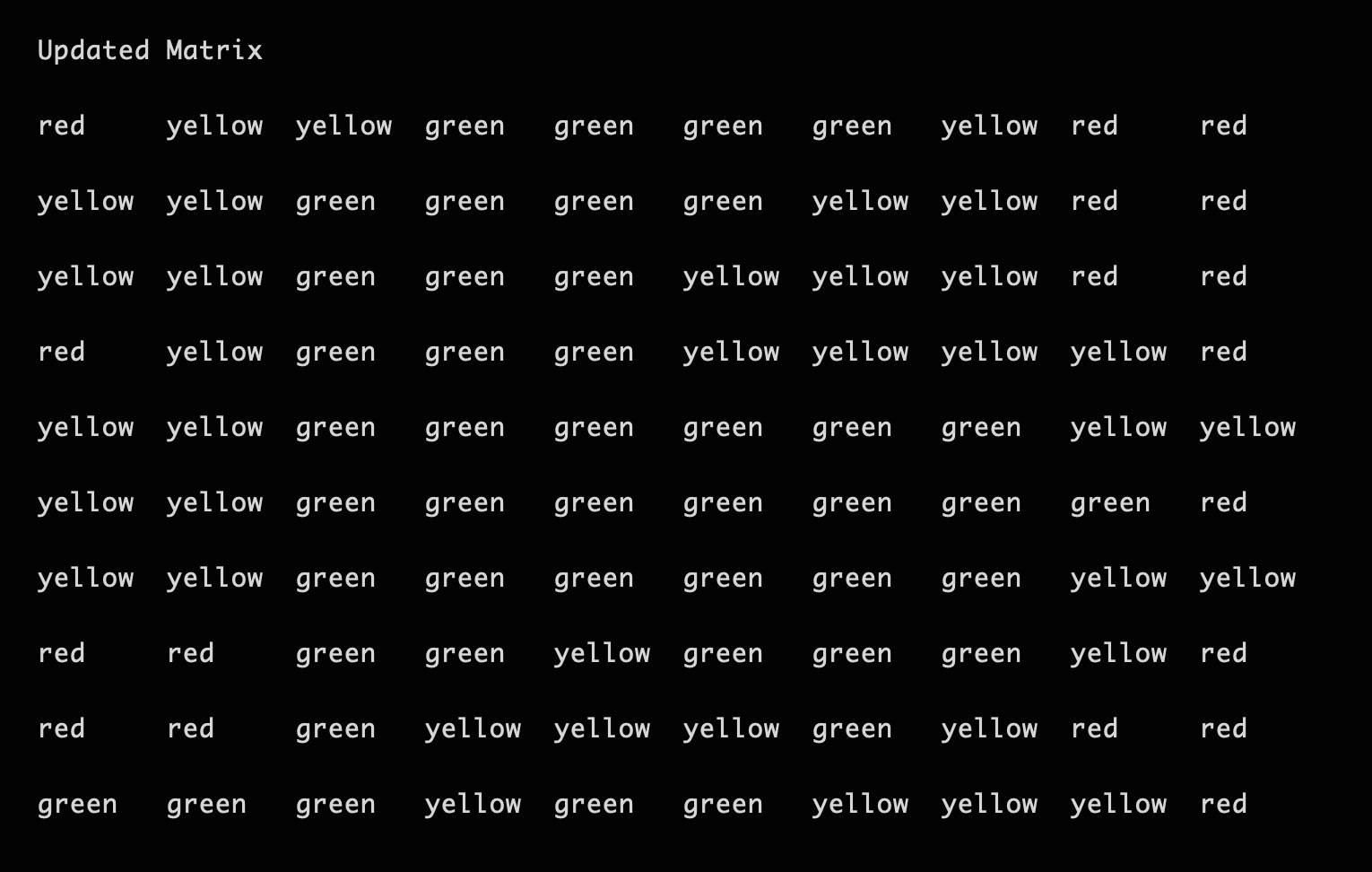
Code Screenshots :

Week 1 :

Here we have taken a grid of 10X10 matrix and representing the all the 100 regions

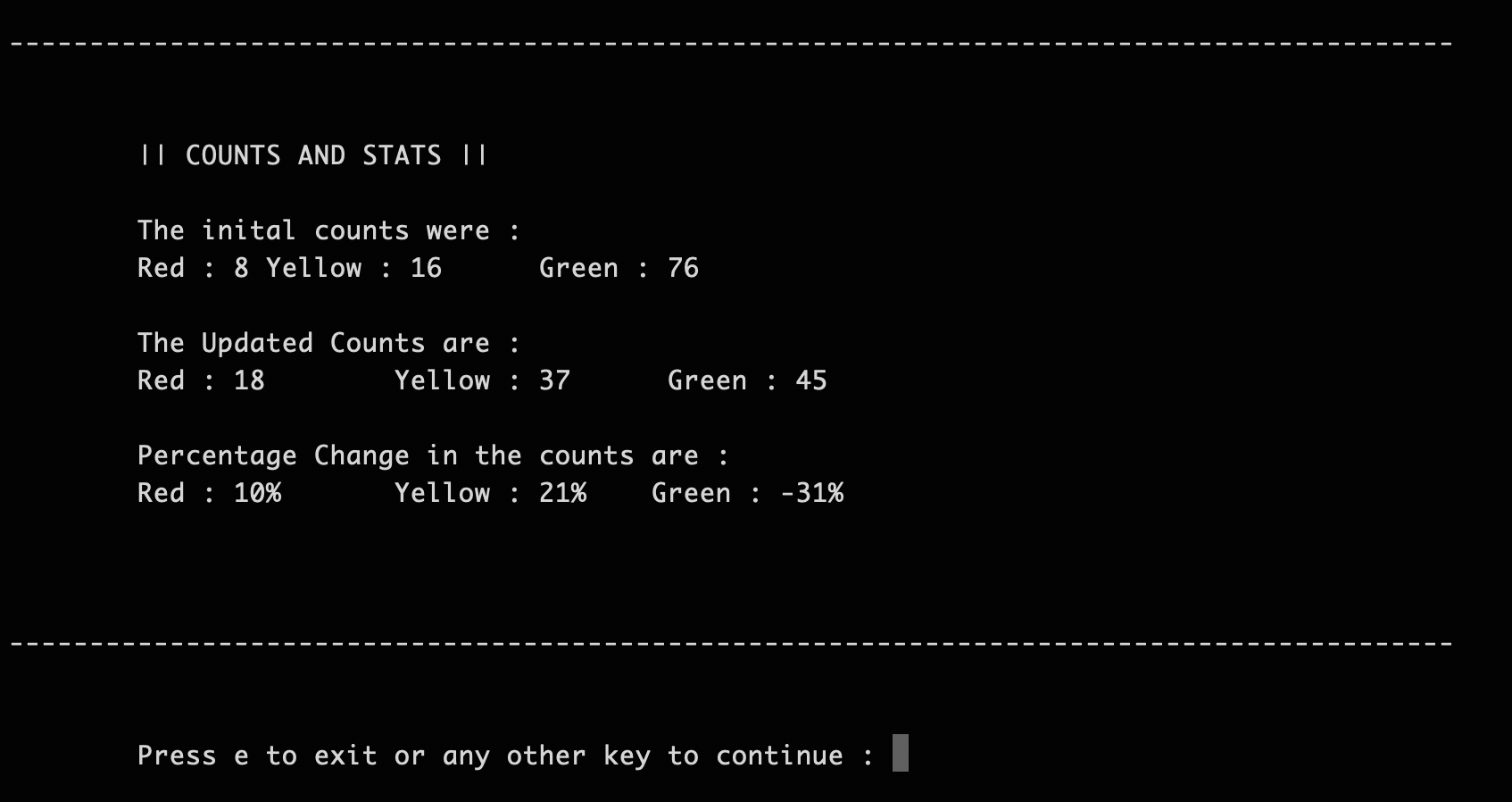


**After Week 1 Stats :**



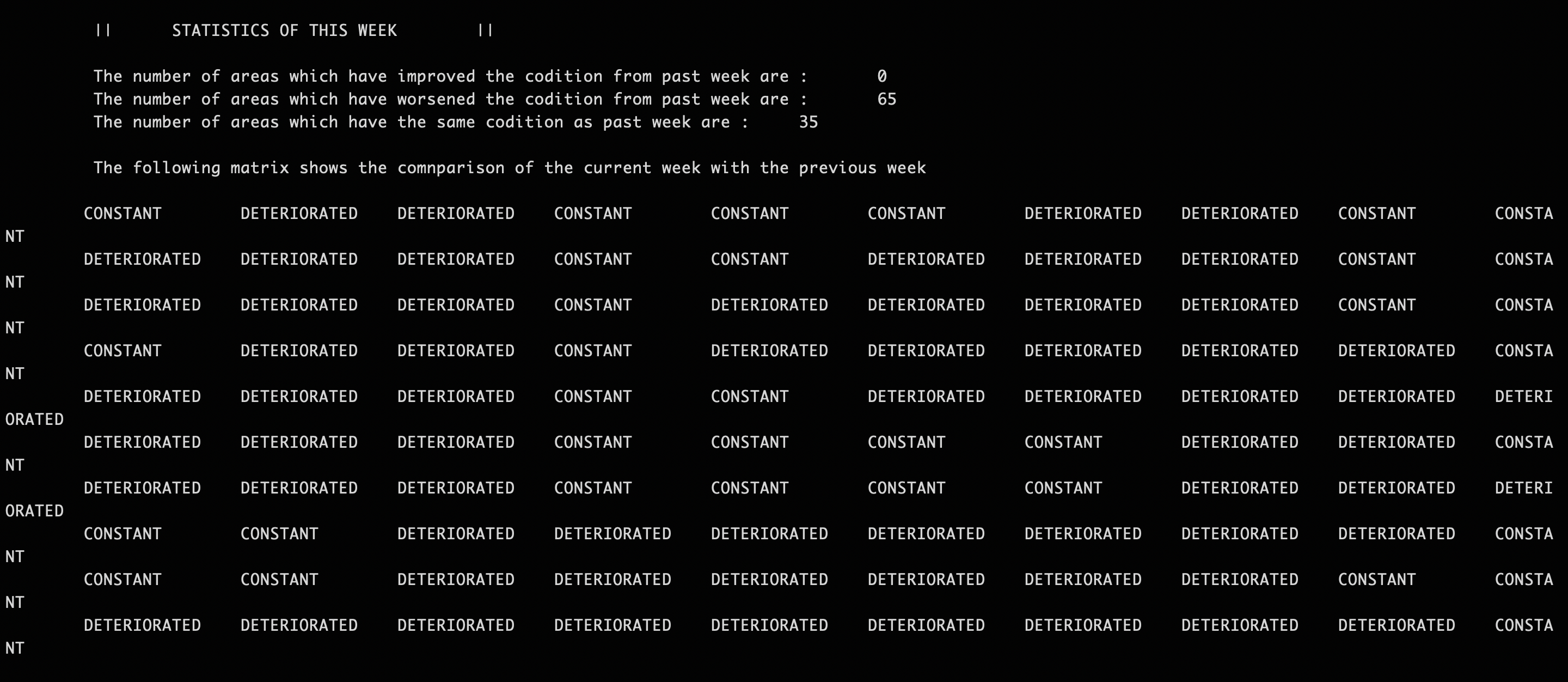
**Stats on the change :**





**Week 2 :**

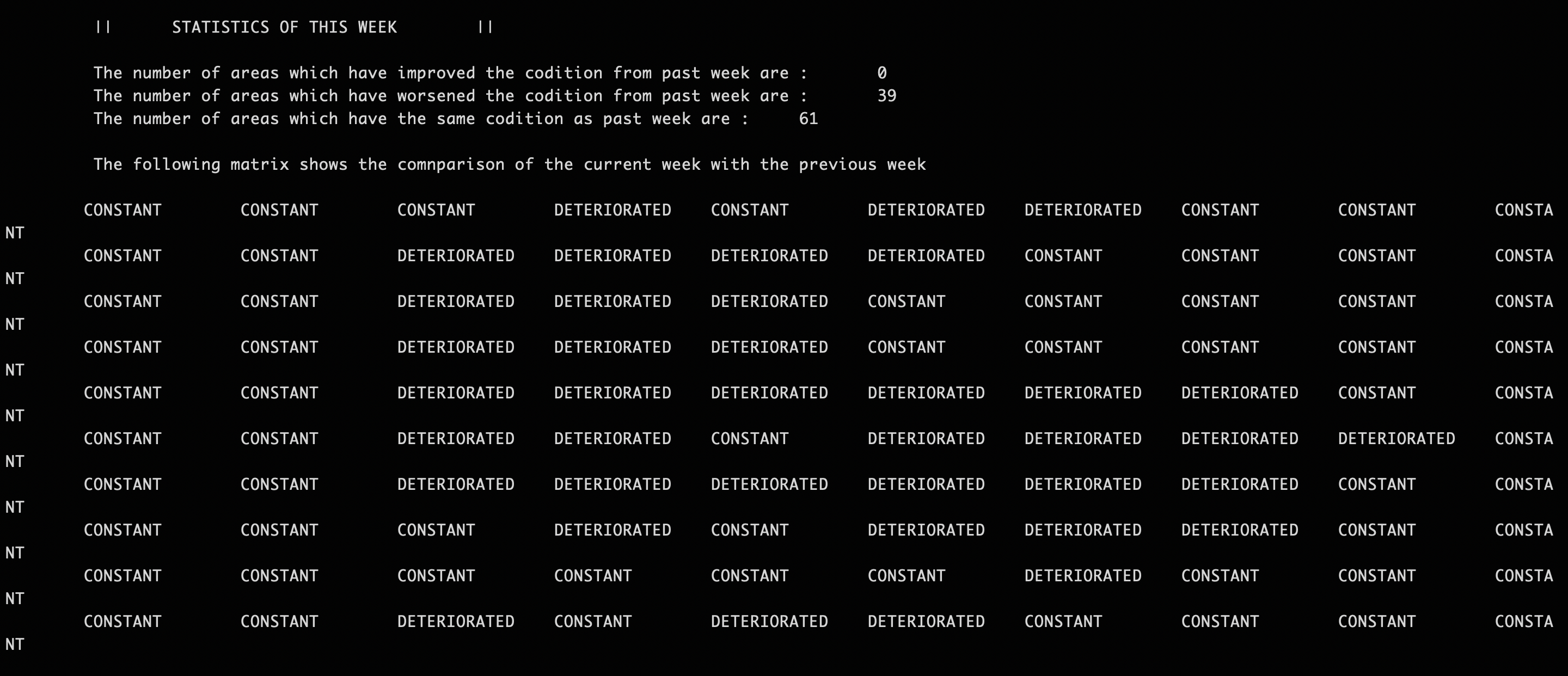






**Week 3:**







**Source Code to the project :**

#include "bits/stdc++.h"

using namespace std;

int rows,cols;

vector<vector<int>> arr;

vector<vector<int>> update;

int fourth\_if(int,int,int);

vector<vector<pair<int,int>>>cnt; // cnt matrix where each element is (colour,cnt)

bool infield( int r, int c )

{

if( r < 0 || r >= rows ) return false;

if( c < 0 || c >= cols ) return false;

return true;

}

// PRINT MATRIX

void print\_Matrix(vector<vector<int>> matrix)

{

unordered\_map<int,string>mp;

mp[1]="red";

mp[2]="yellow";

mp[3]="green";

int rows=matrix.size();

int coloums= matrix[0].size();

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

{

for(int j=0;j<coloums;j++)

{

cout<<"\t"<<mp[matrix[i][j]];

}

cout<<"\n\n";

}

}

int second\_if(int obj, int i, int j)

{

if(obj == 1)

{

return obj;

}

else

{

int a1 = 0,a2 = 0,a3 = 0,a4 = 0,a5 = 0,a6 = 0,a7 = 0,a8 = 0;

if(i>0 and j>0)

a1 = arr[i-1][j-1];

if(j>0)

a2 = arr[i][j-1];

if(j>0 and i<rows-1)

a3 = arr[i+1][j-1];

if(i>0)

a4 = arr[i-1][j];

if(i<rows-1)

a5 = arr[i+1][j];

if(i>0 and j<cols-1)

a6 = arr[i-1][j+1];

if(j<cols-1)

a7 = arr[i][j+1];

if(i<rows-1 and j<cols-1)

a8 = arr[i+1][j+1];

vector<int> vec;

vec.push\_back(a1);

vec.push\_back(a2);

vec.push\_back(a3);

vec.push\_back(a4);

vec.push\_back(a5);

vec.push\_back(a6);

vec.push\_back(a7);

vec.push\_back(a8);

int count1 = count(vec.begin(),vec.end(),1) + count(vec.begin(),vec.end(),2);

if(count1>1)

obj--;

return obj;

}

}

int first\_if(int obj,int i,int j)

{

if(obj==1) //already red

{

return obj;

}

int c=0;

if(infield(i-1,j) && arr[i-1][j]==1) //if red

{

c++;

}

if(infield(i,j-1) && arr[i][j-1]==1)

{

c++;

if(c>=2)

{

return 1;

}

}

if(infield(i-1,j-1) && arr[i-1][j-1]==1 )

{

c++;

if(c>=2)

{

return 1;

}

}

if(infield(i+1,j) && arr[i+1][j]==1 )

{

c++;

if(c>=2)

{

return 1;

}

}

if(infield(i,j+1) && arr[i][j+1]==1 )

{

c++;

if(c>=2)

{

return 1;

}

}

if(infield(i+1,j+1) && arr[i+1][j+1]==1 )

{

c++;

if(c>=2)

{

return 1;

}

}

if(infield(i+1,j-1) && arr[i+1][j-1]==1 )

{

c++;

if(c>=2)

{

return 1;

}

}

if(infield(i-1,j+1) && arr[i-1][j+1]==1 )

{

c++;

if(c>=2)

{

return 1;

}

}

return obj;

}

int fourth\_if(int ele,int i,int j)

{

if(cnt[i][j].second == 4 && cnt[i][j].first < 3)

{

cnt[i][j].first += 1;

cnt[i][j].second = 0;

}

else if(cnt[i][j].first == ele)

{

cnt[i][j].second += 1;

}

else

{

cnt[i][j].first = ele;

cnt[i][j].second = 0;

}

return cnt[i][j].first;

}

void count\_num()

{

int count1\_init=0,count2\_init=0,count3\_init=0;

int count1\_fin=0,count2\_fin=0,count3\_fin=0;

float perc\_1=0,perc\_2=0,perc\_3=0;

for(auto& v: arr)

{

count1\_init+=count(v.begin(),v.end(),1);

count2\_init+=count(v.begin(),v.end(),2);

count3\_init+=count(v.begin(),v.end(),3);

}

for(auto& v: update)

{

count1\_fin+=count(v.begin(),v.end(),1);

count2\_fin+=count(v.begin(),v.end(),2);

count3\_fin+=count(v.begin(),v.end(),3);

}

cout<<"\n\n------------------------------------------------------------------------------------------\n\n";

cout<<endl<<"\t|| COUNTS AND STATS || \n";

cout<<endl<<"\tThe inital counts were : \n";

cout<<"\tRed : "<<count1\_init<<"\tYellow : "<<count2\_init<<"\t Green : "<<count3\_init;

cout<<endl<<"\n\tThe Updated Counts are : \n";

cout<<"\tRed : "<<count1\_fin<<"\tYellow : "<<count2\_fin<<"\t Green : "<<count3\_fin<<endl;

cout<<"\n\tPercentage Change in the counts are : "<<endl;

perc\_1=(((count1\_fin-count1\_init)\*100)/(cols\*rows));

perc\_2=(((count2\_fin-count2\_init)\*100)/(cols\*rows));

perc\_3=(((count3\_fin-count3\_init)\*100)/(cols\*rows));

cout<<"\tRed : "<< perc\_1 << "%"<<"\tYellow : "<<perc\_2<< "%"<<"\tGreen : "<<perc\_3<<"%\n\n"<<endl;

cout<<"\n";

cout<<"------------------------------------------------------------------------------------------\n\n";

}

void weeklyStatistics(vector<vector<int>> Initialmat, vector<vector<int>> Updatedmat)

{

//deterioration

int row = rows;

int col = cols;

vector<vector<string>>stat(rows,vector<string>(col));

int better=0,worse=0,cons=0;

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

{

for(int j=0;j<col;j++)

{

if(Updatedmat[i][j] < Initialmat[i][j])

{

stat[i][j]="DETERIORATED";

worse++;

}

else if(Updatedmat[i][j] > Initialmat[i][j])

{

stat[i][j]="IMPROVED";

better++;

}

else

{

stat[i][j]="CONSTANT";

cons++;

}

}

}

cout<<"\t ||\t STATISTICS OF THIS WEEK \t||\n\n";

cout<<"\t The number of areas which have improved the codition from past week are :\t "<<better<<"\n";

cout<<"\t The number of areas which have worsened the codition from past week are :\t "<<worse<<"\n";

cout<<"\t The number of areas which have the same codition as past week are :\t "<<cons<<"\n";

cout<<"\n";

cout<<"\t The following matrix shows the comnparison of the current week with the previous week \n\n";

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

{

for(int j=0;j<col;j++)

{

cout<<"\t"<<stat[i][j];

}

cout<<"\n";

}

}

// CHECKING THE 4 CONDITIONS FOR EACH ELEMENT

void conditions(vector<vector<int>>v)

{

int row = v.size();

int col = v[0].size();

int new\_obj;

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

{

for(int j=0;j<col;j++)

{

// First if Condn

new\_obj = first\_if(v[i][j],i,j);

if(new\_obj!=v[i][j])

{

update[i][j]=new\_obj;

continue;

}

// Second if Condn

new\_obj = second\_if(v[i][j],i,j);

if(new\_obj!=v[i][j])

{

update[i][j]=new\_obj;

continue;

}

//Third if condition is a subset of second if second so it is covered In that

// Fourth if Condn

new\_obj = fourth\_if(v[i][j],i,j);

if(new\_obj != v[i][j])

{

update[i][j] = new\_obj;

continue;

}

}

}

}

void homepage()

{

cout<<"Enter the length of map : ";

cin>>rows;

cout<<"Enter the breadth of map : ";

cin>>cols;

cout<<"\nThe colour coding for the map is as follows";

cout<<"\n\n";

cout<<"1) Green region is marked as 3 in the map\n";

cout<<"2) Yellow region is marked as 2 in the map\n";

cout<<"3) Red region is marked as 1 in the map \n\n";

cout<<"Enter the intial state of the map as two dimensional array";

cout<<"\n\n";

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

{

vector<int> temp;

vector<int> temp2;

for(int j=0;j<cols;j++)

{

int x;

cin>>x;

temp.push\_back(x);

temp2.push\_back(0);

}

arr.push\_back(temp);

update.push\_back(temp2);

}

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

{

vector<pair<int,int>>t1;

for(int j=0;j<cols;j++)

{

t1.push\_back(make\_pair(arr[i][j],0));

}

cnt.push\_back(t1);

}

update = arr;

while(true)

{

system("clear");

std::this\_thread::sleep\_for(std::chrono::milliseconds(1000));

conditions(arr);

cout<<"\n\n\tUpdated Matrix\n\n";

print\_Matrix(update);

cout<<"\n";

weeklyStatistics(arr,update);

count\_num();

arr = update;

char ch;

cout<<"\n\tPress e to exit or any other key to continue : ";

cin>>ch;

if(ch=='e')

break;

}

}

int main()

{

homepage();

return 0;

} // end of code