Practical 7

Name: Aaron Emmanuel Rocque
Batch: A2 Roll No: 31
Subject: DAA Lab

Problem Statement:

Implement a program that prints all the Hamiltonian cycles present in a connected graph.

Code:

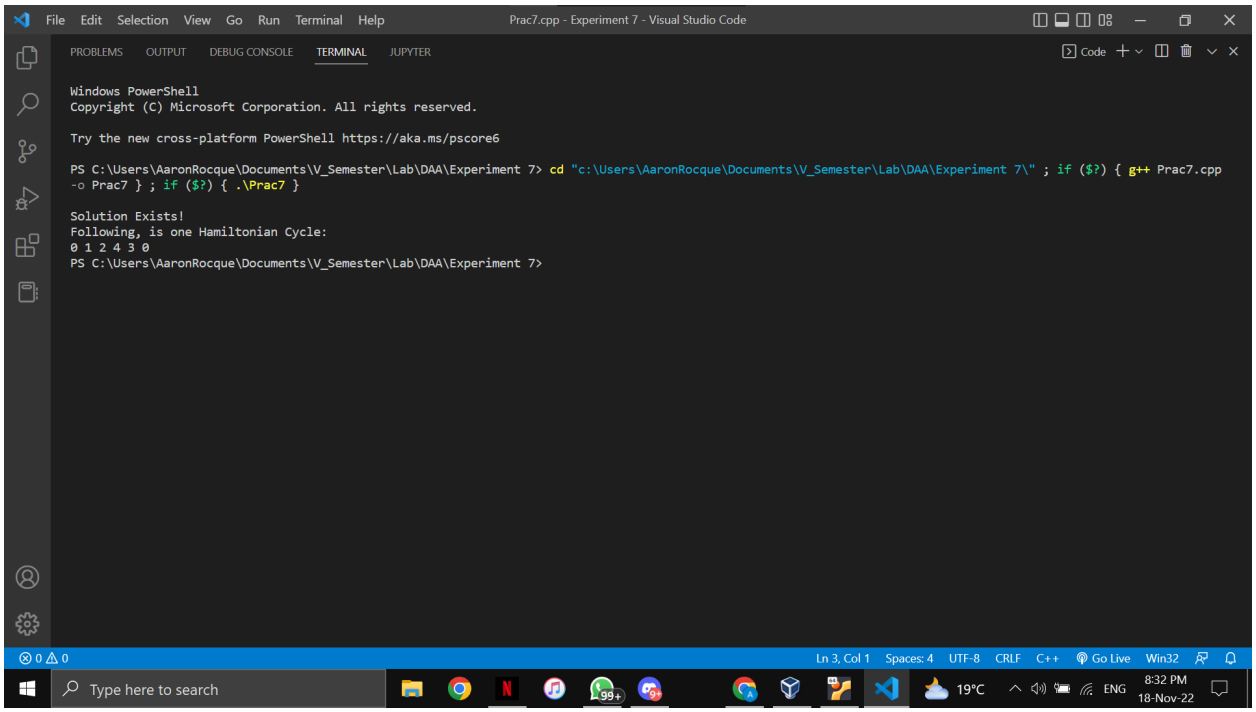
```
#include <bits/stdc++.h>
using namespace std;

#define V 5

bool isSafe(int v, bool graph[V][V], int path[], int pos)
{
    //Check if their is a connection between this vertex and the previously added
one
    if(graph[path[pos-1]][v] == 0)
    {
        return false;
    }

    //Check if the vertex is already in the path array
    for (int i = 0; i < pos; i++)
    {
        if (path[i] == v)
        {
            return false;
        }
    }

    return true;
```

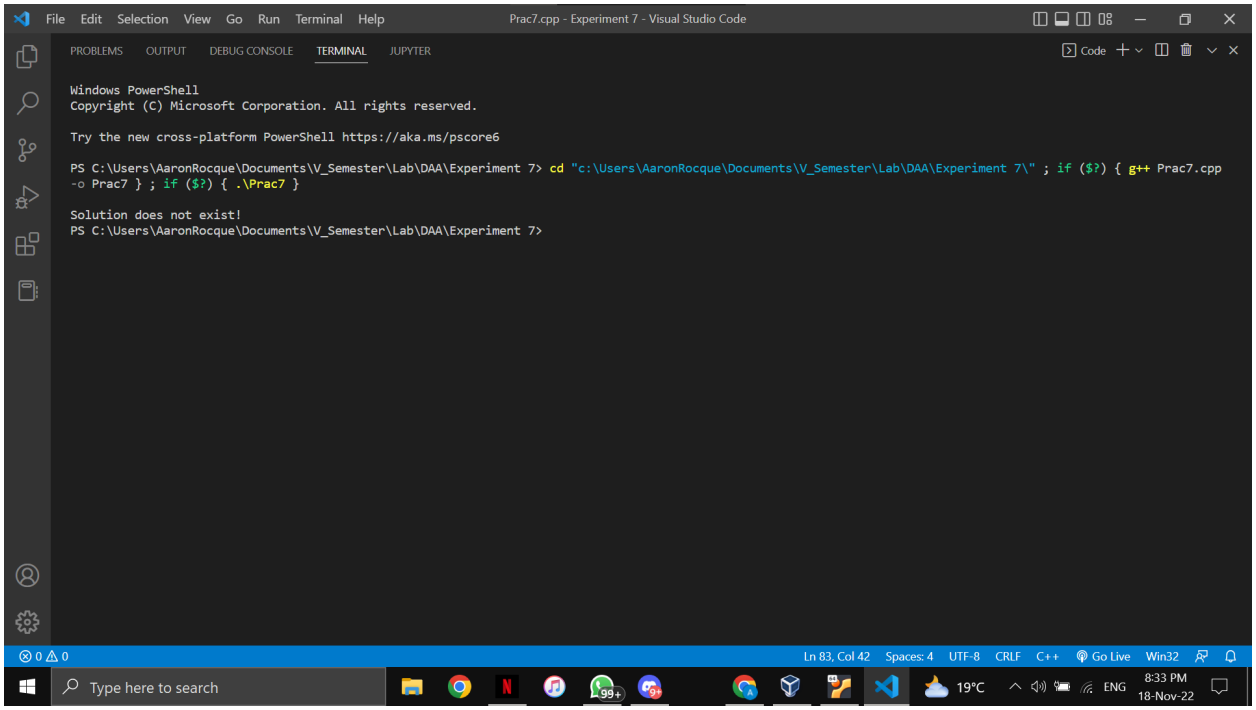
```
}

bool hamCycle(bool G[V][V], int path[], int pos)
{
    //Case 1: All vertices are included in the path
    if(pos == V)
    {
        //This means there is a connection between the first and last vertex
        if(G[path[pos - 1]][path[0]] == 1)
        {
            return true;
        }
        else
        {
            return false;
        }
    }

    //Case 2: Checking different different vertices with their different
different connections
        //Not stating with 0 cause its already selected
    for(int vertex = 1; vertex < V; vertex++)
    {
        if (isSafe(vertex, G, path, pos))
        {
            path[pos] = vertex;

            //Calling the function again with the newly added vertex
            if (hamCycle (G, path, pos + 1) == true)
            {
                return true;
            }

            //If it doesn't return true, means it did not get a new vertex,
            //means the old vertex was a bad choice,
            //we will remove it
            path[pos] = -1;
        }
    }
```

```
    //If we reach here, this means we never returned anything,
    //this means we couldn't make the cycle
    return false;

}

int main()
{
    //CYCLE
    bool graph[V][V] = {{0, 1, 0, 1, 0},
                        {1, 0, 1, 1, 1},
                        {0, 1, 0, 0, 1},
                        {1, 1, 0, 0, 1},
                        {0, 1, 1, 1, 0}};

    //NO CYCLE
    // bool graph[V][V] = {{0, 1, 0, 1, 0},
    //                     {1, 0, 1, 1, 1},
    //                     {0, 1, 0, 0, 1},
    //                     {1, 1, 0, 0, 0},
    //                     {0, 1, 1, 0, 0}};

    //This makes an array 'path' of size V
    int *path = new int[V];

    //Initializing the path with -1
    for(int i = 0; i < V; i++)
    {
        path[i] = -1;
    }

    path[0] = 0;

    if(hamCycle(graph, path, 1) == false)
    {
        cout << "\nSolution does not exist!\n";
    }
    else
```

```
    {
        cout << "\nSolution Exists!\n";
        cout << "Following, is one Hamiltonian Cycle: \n";
        for (int i = 0; i < V; i++)
        {
            cout << path[i] << " ";
        }

        //Printing first vertex again to show a complete cycle
        cout << path[0] << " ";
        cout << endl;
        }
}
```

Output:




File Edit Selection View Go Run Terminal Help
Prac7.cpp - Experiment 7 - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\AaronRocque\Documents\V_Semester\Lab\DAA\Experiment 7> cd "c:\Users\AaronRocque\Documents\V_Semester\Lab\DAA\Experiment 7\" ; if ($?) { g++ Prac7.cpp -o Prac7 } ; if ($?) { .\Prac7 }
Solution does not exist!
PS C:\Users\AaronRocque\Documents\V_Semester\Lab\DAA\Experiment 7>
Ln 83, Col 42 Spaces: 4 UTF-8 CRLF C++ Go Live Win32
Type here to search
19°C
ENG
8:33 PM
18-Nov-22