LAB ASSIGNMENT

AIM :

Write a programs for constraint satisfaction problems like water jug, n-queen problem, crytoarithmatic problem , etc.

Water jug problem

Algorithm

Water Jug problem using BFS

We are given a m litre jug and a n litre jug . Both the jugs are initially empty. The jugs don’t have markings to allow measuring smaller quantities. You have to use the jugs to measure d litres of water where d is less than n.

(X, Y) corresponds to a state where X refers to amount of water in Jug1 and Y refers to amount of water in Jug2

Now, determining the path from initial state (xi, yi) to final state (xf, yf), where (xi, yi) is (0, 0) which indicates both Jugs are initially empty and (xf, yf) indicates a state which could be (0, d) or (d, 0).

The operations you can perform are:

Empty a Jug, (X, Y)->(0, Y) Empty Jug 1

Fill a Jug, (0, 0)->(X, 0) Fill Jug 1

Pour water from one jug to the other until one of the jugs is either empty or full, (X, Y) -> (X-d, Y+d)

Programs in C++

#include<iostream>

#include<iomanip>

#include<math.h>

using namespace std;

int xcapacity;

int ycapacity;

void display(int a, int b,int s);

int min(int d, int f)

{

if (d < f)

return d;

else

return f;

}

int steps(int n)

{

int x = 0, y = 0, step = 0;

int temp;

cout << setw(60) << " Vessel A Vessel B Steps" << endl;

while (x != n )

{

if (x == 0)

{

x = xcapacity;

step += 1;

cout << "Fill X "; display(x, y,step);

}

else if (y == ycapacity)

{

y = 0;

step++;

cout << "Empty Y "; display(x, y,step);

}

else

{

temp = min(ycapacity - y, x);

y = y + temp;

x = x - temp;

step++;

cout << "Pour X in Y"; display(x, y, step);

}

}

return step;

}

void display(int a, int b,int s)

{

cout << setw(16) << a << setw(15) << b << setw(15)<<s<<endl;

}

int main()

{

int n, ans;

cout << "Enter the liters(GOAL) of water required to be filled in Vessel 1:";

cin >> n;

cout << "Enter the capacity of the vessel: ";

cin >> xcapacity;

cout << "Enter the capacity of the second vesssel ";

cin >> ycapacity;

ans = steps(n);

cout << "Steps Required:" << ans;

system("pause");

}

OUTPUT



