# 《离散数学》课程实验报告6 Warshall算法求解传递闭包

## 1.实验原理和方法

对以矩阵表示的关系，使用warshall算法求它的传递闭包，该算法通过逐步更新每个点对之间的可达性来构造传递闭包。基本步骤如下：

1.初始化矩阵 T(0)=AT(0)=A，即将原矩阵 AA 赋给 TT。

2.对于每一对节点 i,ji,j，检查是否存在一个中间节点 kk，使得从 ii 到 jj 可以通过 kk 达到。

3.更新传递关系矩阵：

T(k)[i][j]=T(k−1)[i][j] or (T(k−1)[i][k] and T(k−1)[k][j])T(k)[i][j]=T(k−1)[i][j] or (T(k−1)[i][k] and T(k−1)[k][j])

这里的意思是，如果从 ii 到 kk 可达，并且从 kk 到 jj 可达，那么 ii 到 jj 也可达。

4.重复步骤 2 和 3，直到所有节点都考虑完。

最后，矩阵 TT 就是传递闭包。

## C++源代码

#include <iostream>

#include <vector>

using namespace std;

// 函数声明

vector<vector<bool>> readMatrix(int n);

vector<vector<bool>> computeTransitiveClosure(const vector<vector<bool>>& matrix, int n);

void printMatrix(const vector<vector<bool>>& matrix, int n);

int main() {

int n;

cout << "请输入节点数n: ";

cin >> n;

if (n <= 0) {

cout << "节点数必须大于0。" << endl;

return 1;

}

vector<vector<bool>> matrix = readMatrix(n);

vector<vector<bool>> closure = computeTransitiveClosure(matrix, n);

cout << "传递闭包矩阵:" << endl;

printMatrix(closure, n);

return 0;

}

// 读取邻接矩阵

vector<vector<bool>> readMatrix(int n) {

vector<vector<bool>> matrix(n, vector<bool>(n));

cout << "请输入邻接矩阵（每行n个0或1，以空格分隔）:" << endl;

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

for (int j = 0; j < n; ++j) {

int input;

cin >> input;

if (cin.fail() || (input != 0 && input != 1)) {

cout << "输入错误，请输入0或1。" << endl;

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

return readMatrix(n); // 递归重新读取矩阵

}

matrix[i][j] = bool(input);

}

}

return matrix;

}

// 计算传递闭包

vector<vector<bool>> computeTransitiveClosure(const vector<vector<bool>>& matrix, int n) {

vector<vector<bool>> closure(matrix);

// Warshall算法更新可达性矩阵

for (int k = 0; k < n; ++k) {

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

for (int j = 0; j < n; ++j) {

closure[i][j] = closure[i][j] || (closure[i][k] && closure[k][j]);

}

}

}

return closure;

}

// 输出矩阵

void printMatrix(const vector<vector<bool>>& matrix, int n) {

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

for (int j = 0; j < n; ++j) {

cout << matrix[i][j] << " ";

}

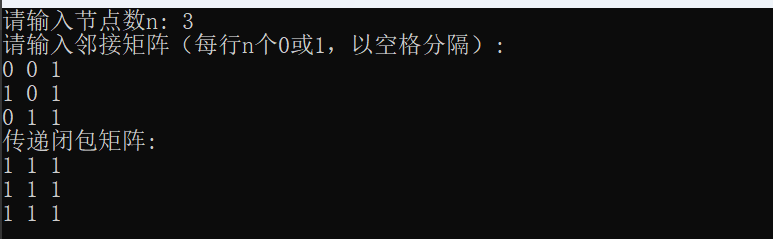
cout << endl;

}

}

## 3．实验结果

求关系的传递闭包：



## 4．实验小结

通过这道题目，我实现了使用c++代码和warshall算法求矩阵的传递闭包。与之前的作业中使用公式Mt=M+M^2+M^3+…求解传递闭包相比，warshall算法会简便很多，理解起来也比较容易，这让我感受到了warshall算法的优越性。