华南理工大学

《课程名称》课程实验报告

实验题目：The Prim Algorithm for MST

姓名： 谭演锋 学号： 202130100456

班级： 计算机全英联合班 组别： 无

合作者： 无

指导教师： 颜小洋

|  |
| --- |
| **实验概述** |
| 【实验目的及要求】  实验目的：  Solve Minimum Spanning Tree (MST) Problem by Greedy Algorithms——using Kruskal’s algorithm and Prim’s algorithm. Compare the differences between two algorithms.  实验要求：  Given a undirected Graph G= (V, E) like below, to calculate the minimum spanning tree using Kruskal’s algorithm and Prim’s algorithm. The template should be used for all kinds of data type, such as: integer, real, double, etc. in the program. Programs should be made using Object-Oriented Programming (OOP) method. Write down the report in which there should be the execution results of the program. There should be the selection processes and their differences between two algorithms, besides the final MST for the given graph in the report.  The template should be used for all kinds of data type, such as: integer, real, double, etc. in the program.  Programs should be made by Object-Oriented Programming (OOP) method.  The results should be compared with ones of other algorithms, such as: Straight selection sort, insert sort, etc., and draw the graph to find their differences.  Write down the report in which there should be the execution results of the program.  【实验环境】  操作系统：Windows XP |
| **。** |
| 【实验过程】   1. 实验步骤： 2. Definition of class and function：   **Class:** MST\_Alg  **Data member:**  Graph:in the form of adjacency matrix  **Member functions:**  minKey: return the min\_index of the graph  printMST: print the process of construct minimum spanning tree  primMST：Use Prim’s algorithm to construct the minimum spanning tree  findMin: find the next minimum edge and union two trees (used for Kruskal algorithm)  KruskalMST: Use Kruskal’s algorithm to construct the minimum spanning tree  **Class:** UF  **Data member:**  count: Number of connected node  size: A tree’s “weight”  parent: Recording a tree  **Member functions:**  Union: Connect two nodes into one tree and balance them  connected: Evaluate whether two nodes are connected  find: Return the parent node and compress the route  Count: Return the data member count  Class UF is used for the Kruskal’s algorithm to construct the minimum spanning tree.   1. Run program in the main function:   Initial the graph and implement the Prim’s algorithm and Kruskal’s algorithm. And use the clock() to calculate the running time of the function.   1. 实验数据：   The graph used in the experiment is followed：     1. 实验主要过程：   There are 2 main procedures of the experiment Prim’s algorithm and Kruskal’s algorithm:  Prim:  Input: A weighted, connected, and undirected graph G = (V, E).  Output: A minimum spanning tree for G.  step1. Let x be any vertex in V. Let X={x} and Y=V-{x}. step2. Select an edge (u, v) from E such that u ∊ X, v ∊ Y and (u, v) has the smallest weight among edges between X and Y. step 3. Connect u to v. Let X= X∪{v} and Y= Y- {v}.  step 4. If Y is empty, terminate and the resulting tree is a minimum spanning tree. Otherwise, Go to step2.  Kruskal:  Input: A weighted, connected, and undirected graph G = (V, E).  Output: A minimum spanning tree for G.  T: = 𝟇  While T contains less than n - edges do  Begin  Choose an edge (v w) from E of the smallest weight.  Delete (v, w) from E  If (the adding of (v, w) to T does not create a cycle in T) then.  Add (v, w) to T  Else  Discard (v, w)  End   1. 实验结果：   表格  描述已自动生成  We can know that:  The minimal weight of the graph is 105.  Prim’s algorithm is faster than Kruskal’s algorithm since it is a dense graph. Time complexity of Prim’s algorithm is O(n2) and Kruskal’s algorithm is O(ElogE) where n is the number of nodes and E is the number of edges.  According to the program we can get the MST constructing procedure of the Prim’s algorithm:    The MST constructing procedure of the Prim’s algorithm: |
| **小结** |
| After the experiment, I was more familiar with the details of the Prim’s algorithm and the Kruskal’s algorithm. Not just the pseudocode written on the book. I have known how to create a set to store the nodes visited, and according to the visited nodes set to select the next expand edge. Additionally, by writing the Kruskal’s algorithm I have also known what the Union and Find set is. It is used to find whether two set that I find in the Kruskal’s algorithm will become a circle if I connect them. Besides I have learned the difference between the Prim’s algorithm and the Kruskal’s algorithm. Prim’s algorithm use greedy method by selecting the edge that can be expanded by current expanded nodes and has the smallest weight. It runs n times to ensure we select n-1 edges to connect n nodes. Each iteration selects one edge to connect a new node. However, the Kruskal’s algorithm use greedy method by selecting the edges with the lowest weight and exam whether the edge does not construct a cycle with the selected edges. If the edge constructs a circle, we discard it. Otherwise, we add it to the set of selected edges. We use a while loop to implement it until the edges set has n-1 edges. All the two algorithms can get a minimal spanning tree. Prim’s algorithm is faster than Kruskal’s algorithm since it is a dense graph. Time complexity of Prim’s algorithm is O(n2) and Kruskal’s algorithm is O(ElogE) where n is the number of nodes and E is the number of edges. |
| **指导教师评语及成绩** |
| 评语：  成绩：           指导教师签名：                                                 批阅日期： |