学习情况表

|  |  |  |  |
| --- | --- | --- | --- |
| **姓名** | 张蕊晗 | **学号** | 2020905198 |
| **学院** | 信息工程学院 | **专业** | 电子信息专业 |

（请在下面表格本周学习情况）

|  |
| --- |
| 学习情况简述 |
| 贪婪算法，Dijkstra算法，Kruskal算法 |
| 本周练习过的代码（例） |
| （本周练习写过的代码，如实填写，疑似抄袭会单独测试）  *代码1：*  *#include<stdio.h>*  *int main()*  *{*  *printf("HelloWorld\n");*  *return 0;*  *}*  *代码2：*  …….. |
| 本周练习过的代码 |
| #include<iostream>  #include<algorithm>  #define INF 1000  #define MAX\_V 100  using namespace std;    int main()  {  int V,E;  int i,j,m,n;  int cost[MAX\_V][MAX\_V];  int d[MAX\_V];  bool used[MAX\_V];  cin>>V>>E;  fill(d,d+V+1,INF);  fill(used,used+V,false);  for(i=0;i<V;i++)  {  for(j=0;j<V;j++)  {  if(i==j) cost[i][j]=0;  else cost[i][j]=INF;  }  }  for(m=0;m<E;m++)  {  cin>>i>>j>>cost[i][j];  cost[j][i]=cost[i][j];  }  cin>>n;  d[n]=0;  //源点  while(true)  {  int v=V;  for(m=0;m<V;m++)  {  if((!used[m])&&(d[m]<d[v])) v=m;  }  if(v==V) break;  used[v]=true;  for(m=0;m<V;m++)  {  d[m]=min(d[m],d[v]+cost[v][m]);  }  }  for(i=0;i<V;i++)  {  cout<<"the shortest distance between "<<n<<" and "<<i<<" is "<<d[i]<<endl;  }  }  #include<iostream>  #include<algorithm>  #define MAX\_E 100  using namespace std;  struct edge  {  int u,v,cost;  };  int pre[MAX\_E];  edge es[MAX\_E];  int find(int x);  void initvalue(int x);  bool same(int x,int y);  void unite(int x,int y);  bool comp(const edge& e1,const edge& e2);    int main()  {  int V,E;  int i,j,m,n;  cin>>V>>E;  initvalue(V);  for(i=0;i<E;i++) cin>>es[i].u>>es[i].v>>es[i].cost;  sort(es,es+E,comp);  int res=0;  for(i=0;i<E;i++)  {  edge e=es[i];  if(!same(e.u,e.v))  {  unite(e.u,e.v);  res+=e.cost;  }  }  cout<<res<<endl;  }    bool comp(const edge& e1,const edge& e2)  {  return e1.cost<e2.cost;  }    void initvalue(int x)  {  for(int i=0;i<x;i++) pre[i]=i;  }    int find(int x)  {  int r=x;  while(pre[r]!=r) r=pre[r];  int i=x,j;  while(pre[i]!=r)  {  j=pre[i];  pre[i]=r;  i=j;  }  return r;  }    bool same(int x,int y)  {  if(find(x)==find(y)) return true;  else return false;  }    void unite(int x,int y)  {  int fx=find(x);  int fy=find(y);  if(fx!=fy) pre[fx]=fy;  }  package com.qf.greedy;  import java.util.\*;  public class GreedyAlgorithm {  public static void main(String[] args) {  /\*  \* 广播台 覆盖地区  K1 "北京", "上海", "天津"  K2 "广州", "北京", "深圳"  K3 "成都", "上海", "杭州"  K4 "上海", "天津"  K5 "杭州", "大连"\*/  //新建广播台列表  Map<String, HashSet<String>> broadCasts=new HashMap<>();  HashSet<String> hashSet1=new HashSet<>();  hashSet1.add("北京");  hashSet1.add("上海");  hashSet1.add("天津");  HashSet<String> hashSet2=new HashSet<>();  hashSet2.add("广州");  hashSet2.add("北京");  hashSet2.add("深圳");  HashSet<String> hashSet3=new HashSet<>();  hashSet3.add("成都");  hashSet3.add("上海");  hashSet3.add("杭州");  HashSet<String> hashSet4=new HashSet<>();  hashSet4.add("上海");  hashSet4.add("天津");  HashSet<String> hashSet5=new HashSet<>();  hashSet5.add("杭州");  hashSet5.add("大连");  broadCasts.put("k1",hashSet1);  broadCasts.put("k2",hashSet2);  broadCasts.put("k3",hashSet3);  broadCasts.put("k4",hashSet4);  broadCasts.put("k5",hashSet5);  //得出所有地区  HashSet<String> allAreas=new HashSet<>();  Set<String> strings = broadCasts.keySet();  for (String string : strings) {  HashSet<String> hashSet = broadCasts.get(string);  for (String city : hashSet) {  allAreas.add(city);  }  }  System.out.println(allAreas);  //所取得结果keyList  List<String> keyList=new ArrayList<>();  //最多覆盖地区的key  String maxKey=null;  //临时储存变量  HashSet<String> tempSet=new HashSet<>();  //循环所有地区的set，有覆盖的地区就从set中删除，直到为空为止  while (allAreas.size()>0){  Set<String> sets = broadCasts.keySet();  //重新设置key为空  maxKey=null;  for (String key : sets) {  tempSet.clear();  HashSet<String> hashSet = broadCasts.get(key);  tempSet.addAll(hashSet);  tempSet.retainAll(allAreas);    //所覆盖的地区size不能为空，覆盖地区大于之前的key，更新  if (tempSet.size()>0&&(maxKey==null||tempSet.size()>broadCasts.get(maxKey).size())){  maxKey=key;  }  }  if (maxKey!=null){  keyList.add(maxKey);  allAreas.removeAll(broadCasts.get(maxKey));  }  }  System.out.println("keyList="+keyList);  }  } |

1. 该表信息将会作为你报名申请的重要依据，请认真仔细填写。
2. 培训班有严格的制度，请认真阅读规则并结合自身情况填写该表。
3. 完成该表填写后以“xxx学习情况”命名，及时上传作业。

最后希望大家能够加入我们，一起努力，共同进步！