

最小生成树之kruskal算法计算机等级考试 PDF转换可能丢失
图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022__E6_9C_80_E5_B0_8F_E7_94_9F_E6_c97_645871.htm 1. Kruskal算法 (1) 算法思想 Kruskal算法每次选择 $n-1$ 条边，所使用的贪婪准则是：从剩下的边中选择一条不会产生环路的具有最小耗费的边加入已选择的边的集合中。注意到所选取的边若产生环路则不可能形成一棵生成树。Kruskal算法分 e 步，其中 e 是网络中边的数目。按耗费递增的顺序来考虑这 e 条边，每次考虑一条边。当考虑某条边时，若将其加入到已选边的集合中会出现环路，则将其抛弃，否则，将它选入。初始时没有任何边被选择。边 $(1, 6)$ 是最先选入的边，它被加入到欲构建的生成树中，得到图13-12c。下一步选择边 $(3, 4)$ 并将其加入树中（如图13-12d所示）。然后考虑边 $(2, 7)$ ，将它加入树中并不会产生环路，于是便得到图13-12e。下一步考虑边 $(2, 3)$ 并将其加入树中（如图13-12f所示）。在其余还未考虑的边中， $(7, 4)$ 具有最小耗费，因此先考虑它，将它加入正在创建的树中会产生环路，所以将其丢弃。此后将边 $(5, 4)$ 加入树中，得到的树如图13-12g所示。下一步考虑边 $(7, 5)$ ，由于会产生环路，将其丢弃。最后考虑边 $(6, 5)$ 并将其加入树中，产生了一棵生成树，其耗费为99。图1-13给出了Kruskal算法的伪代码。(2)C代码

```
/*
Kruskal.c Copyright (c) 2002, 2006 by ctu_85 All Rights Reserved.
*/
/* I am sorry to say that the situation of unconnected graph is not
concerned */
#include "stdio.h"
#define maxver 10
#define maxright 100
int G[maxver][maxver],record=0,touched[maxver][maxver].
```

```
int circle=0. int FindCircle(int,int,int,int). int main() { int
path[maxver][2],used[maxver][maxver]. int
i,j,k,t,min=maxright,exsit=0. int v1,v2,num,temp,status=0. restart:
printf("Please enter the number of vertex(s) in the graph:\n").
scanf("%d", 100Test 下载频道开通 , 各类考试题目直接下载。
详细请访问 www.100test.com
```