

贪心算法

讲授者 王爱娟

1录

贪心法的思想

最小生成树 之PRIM算治

最小生成树 之Kruskal 注

贪心算法

讲授者 王爱娟

aijuan 321 @ fox mail.com

重庆理工大学 计算机科学与工程学院

August 22, 2024

目录

贪心法的思想

敢小生成树 之PRIM算:

最小生成树 之Kruskal舞 法 ① 贪心法的思想

② 最小生成树之PRIM算法

3 最小生成树之Kruskal算法



贪心法的实例

贪心算法

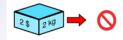
讲授者 王爱娟

选择1:

按物品价值





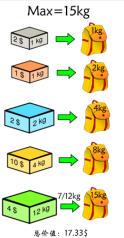




总价值: 13.67\$

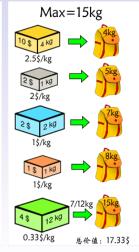
选择2:

按物品重量



选择3:

按物品价值重量比





贪心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算?

取小王成例 之Kruskal算 法 找零问题:

(1) 硬币面额: $d_1 = 25, d_2 = 10, d_3 = 5, d_4 = 1$



贪心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算?

敢小生成树 之Kruskal昇 法 找零问题:

(1) 硬币面额: $d_1 = 25, d_2 = 10, d_3 = 5, d_4 = 1$ 找零48美分?

贪心算法

讲授者 王爱娟

目录 贪心法的思考

最小生成树 之PRIM算法 最小生成树 找零问题:

(1) 硬币面额: $d_1 = 25, d_2 = 10, d_3 = 5, d_4 = 1$ 找零48美分?

 $1 \uparrow d_1 = 25, \ 2 \uparrow d_2 = 10, \ 3 \uparrow d_4 = 1$

选择策略:从当前的几个选择中确定一个最佳选择。

贪心法的思想

在问题的求解过程中

- 可行性: 所做的每一步选择必须满足问题的约束。
- 局部最优:选择不从全局考虑,仅考虑当前状态(局部 状态)。总是做出在**当前看来**是最好的选择
- **不可取消**:选择具备不可取消性,即一旦选择,在算法的后面步骤中就无法改变了!

贪心算法

讲授者 王爱娟

1录

含心法的思想

最小生成树

ZPRIM罪》

之Kruskal算 法 (2) 硬币面额: $d_1 = 25, d_2 = 10, d_3 = 1$



资化界法

讲授者 王爱娟

1录

含心法的思

最小生成树

最小生成树

之Kruskal葬 法 (2) 硬币面额: $d_1 = 25, d_2 = 10, d_3 = 1$ 找零30美分?



讲授者 王受娟

目录

含心法的思想

最小生成树 ラPRIM 質:

最小生成树 之Kruskal箅 (2) 硬币面额: $d_1 = 25, d_2 = 10, d_3 = 1$ 找零30美分?

 $1 \uparrow d_1 = 25, 5 \uparrow d_3 = 1$



贪心芽法

讲授者 王爱娟

1录

贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal算 法 (2) 硬币面额: $d_1 = 25, d_2 = 10, d_3 = 1$ 找零30美分?

 $1 \uparrow d_1 = 25, 5 \uparrow d_3 = 1$

贪心法的特点

贪心算法不是对所有问题都能得到整体最优解,但对范围相 当广泛的许多问题它能产生整体最优解或者是整体最优解的 近似解。



贪心芽法

讲授者 王受娟

目录

贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal斜 法 (2) 硬币面额: $d_1 = 25, d_2 = 10, d_3 = 1$ 找零30美分?

 $1 \uparrow d_1 = 25, 5 \uparrow d_3 = 1$

贪心法的特点

贪心算法不是对所有问题都能得到整体最优解,但对范围相 当广泛的许多问题它能产生整体最优解或者是整体最优解的 近似解。

如何证明某一贪婪算法能获得最优解? (本课程不涉及)



贪心芽法

讲授者 王爱娟

目录

贪心法的思考 ロール エココ

之PRIM算治 最小生成树 之Kruskal算

贪心法的特点

贪心算法不是对所有问题都能得到整体最优解,但对范围相 当广泛的许多问题它能产生整体最优解或者是整体最优解的 近似解。

如何证明某一贪婪算法能获得最优解? (本课程不涉及) 其方法:

- 数学归纳法证明算法每一步获得的部分解能够扩展到全局最优解;
- 证明算法在接近目标的过程中,每一步的选择不会比其 他算法差;
- 基于算法的输出, 证明算法得到的解的最优性;



贪心算法

讲授者 王爱娟

目录

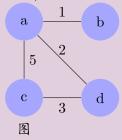
贪心法的思想

最小生成树 之PRIM算》

最小生成树 之Kruskal箅 法

生成树

连通图的一棵<mark>生成树</mark>是包含图的<mark>所有点</mark>的连通无环子图(一棵树)



最小生成树

加权连通图的一棵最小生成树是图的一棵权重最小的生成树

贪心算法

讲授者 王爱娟

目录

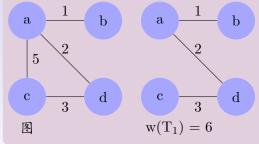
贪心法的思想

取小生成树 之PRIM算法

取小王成俩 之Kruskal算 法

生成树

连通图的一棵<mark>生成树</mark>是包含图的<mark>所有点</mark>的连通无环子图(一棵树)



最小生成树

加权连通图的一棵最小生成树是图的一棵权重最小的生成树

贪心算法

讲授者 王受娟

目录

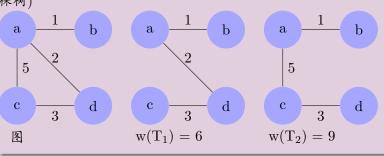
贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal算 法

生成树

连通图的一棵<mark>生成树</mark>是包含图的<mark>所有点</mark>的连通无环子图(一棵树)



贪心算法

讲授者 王爱娟

目录

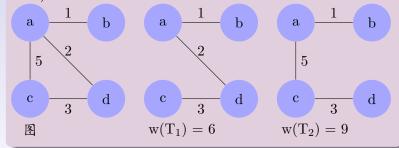
贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal算 法

生成树

连通图的一棵<mark>生成树</mark>是包含图的<mark>所有点</mark>的连通无环子图(一棵树)



最小生成树

加权连通图的一棵最小生成树是图的一棵权重最小的生成树



近代开妆

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

敢小生成树 之Kruskal算 ^汪 回顾:图的定义

一个图G可以用二元组 $\langle V,E \rangle$ 表示,其V表示图的顶点集合。E表示连接顶点的边集



贪心芽法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal箅 艹

回顾:图的定义

一个图G可以用二元组<V,E>表示, 其V表示图的顶点集合, E表示连接顶点的边集

- ❶ 开始产生最小生成树:从V中任意顶点v₀
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和 $V V_T$
 - V_T 表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V V_T$ 。将 $b \in V_T$,并 $d \in V_T$ 0,并 $d \in V_T$ 0,是 $d \in V_T$ 0。是 $d \in V_T$
- ③ 反复执行第2步,直至集合V-V_T为空。



贪心芽法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal算 法

回顾:图的定义

一个图G可以用二元组<V,E>表示, 其V表示图的顶点集合, E表示连接顶点的边集

- 开始产生最小生成树:从V中任意顶点v₀
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和 $V V_T$
 - V_T 表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V V_T$ 。将 $b \to V_T$,并从 $V V_T$ 中删除
- ③ 反复执行第2步,直至集合V-V_T为空。



贪心芽法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal昇 法

回顾:图的定义

一个图G可以用二元组<V,E>表示,其V表示图的顶点集合,E表示连接顶点的边集

- 开始产生最小生成树:从V中任意顶点v₀
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和 $V V_T$
 - V_T表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V V_T$ 。将 $b \to V_T$,并从 $V V_T$ 中删除
- ③ 反复执行第2步,直至集合V-V_T为空。



贪心芽法

讲授者 王愛娟

目录 贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal争 法

回顾:图的定义

一个图G可以用二元组<V,E>表示,其V表示图的顶点集合,E表示连接顶点的边集

- ❶ 开始产生最小生成树:从V中任意顶点v₀
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和 $V V_T$
 - \bullet V_T 表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V V_T$ 。将 $b \in V V_T$,并 $d \in V_T$,在 $d \in V_T$,在d
- ③ 反复执行第2步,直至集合V-V_T为空。



贪心芽法

讲授者 王爱娟

目录 贪心法的思想

最小生成树 之PRIM算法 最小生成树

最小生成树 之Kruskal 法

回顾:图的定义

一个图G可以用二元组<V,E>表示,其V表示图的顶点集合,E表示连接顶点的边集

- ❶ 开始产生最小生成树:从V中任意顶点v₀
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和 $V V_T$
 - \bullet V_T 表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V V_T$ 。将 $b \in V V_T$,并 $d \in V_T$,在 $d \in V_T$,在d
- ◎ 反复执行第2步,直至集合V-V_T为空。



贪心算法

讲授者 王爱娟

目录 会心注的思构

最小生成树 之PRIM算法

最小生成树 之Kruskal算 法

回顾:图的定义

一个图G可以用二元组<V,E>表示,其V表示图的顶点集合,E表示连接顶点的边集

Prim算法概述

- ❶ 开始产生最小生成树:从V中任意顶点v₀
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和 $V V_T$
 - \bullet V_T 表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V V_T$ 。将 $b \to V_T$,并从 $V V_T$ 中删除
- ⑤ 反复执行第2步,直至集合V-V_T为空。

为什么终止条件是 $V - V_T$ 为空

最小生成树的顶点与图的顶点相同! Jump to definition



贪心算法

讲授者 王爱娟

目录

会心法的思

12 1. d. J. 14

東小王成例 之PRIM算法

敢小生成树 之Kruskal算 为什么称Prim算法为贪心算法

每步总是只选择当前最优的解: 最短边!

贪心算法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

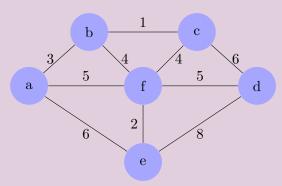
最小生成树 之Kruskal算 斗

为什么称Prim算法为贪心算法

每步总是只选择当前最优的解: 最短边!

Prim算法实例

考虑图G如下所示:



贝代研究

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

取小生成树 之Kruskal算 ヰ

为什么称Prim算法为贪心算法

每步总是只选择当前最优的解: 最短边!

Prim算法实例

考虑图G如下所示:

随机选择一个开始点 b 1 c 3 4 4 6 6 d d 2 8

e

M. Carro

讲授者 王爱娟

目录 会小让的田:

贪心法的思想

最小生成树 之PRIM算法

敢小生成树 之Kruskal算 ^汪

为什么称Prim算法为贪心算法

每步总是只选择当前最优的解: 最短边!

Prim算法实例

考虑图G如下所示:

随机选择一个开始点 b 1 c d 4 5 d d e e

贪心芽法

讲授者 王受娟

目录 含心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal算 斗

为什么称Prim算法为贪心算法

每步总是只选择当前最优的解: 最短边!

b

5

Prim算法实例

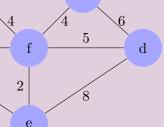
考虑图G如下所示:

有尽图G如下所示: 随机选择一个开始点

 $V_T = \{a\}$

 $V - V_T = \{b,c,d,e,f\}$

a



贪心芽法

讲授者 王爱娟

目录 贪心法的思想

更心法的心思

取小生成树 之PRIM算法

取小生成柄 之Kruskal算 ^注

为什么称Prim算法为贪心算法

每步总是只选择当前最优的解: 最短边!

Prim算法实例

考虑图G如下所示:

随机选择一个开始点 b 1 c $V_T = \{a\}$ a b f d $V - V_T = \{b,c,d,e,f\}$ b d e

贪心芽法

讲授者 王受娟

目录 贪心法的思考

最小生成树 之PRIM算法

取小生成树 之Kruskal算 ヰ

为什么称Prim算法为贪心算法

每步总是只选择当前最优的解: 最短边!

Prim算法实例

考虑图G如下所示:

随机选择一个开始点 $\frac{1}{5}$ $\frac{1}{6}$ $\frac{1$

贪心芽法 讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

取小王成例 之Kruskal算 注

为什么称Prim算法为贪心算法

每步总是只选择当前最优的解: 最短边!

Prim算法实例

考虑图G如下所示:

随机选择一个开始点 $\frac{1}{5}$ $\frac{1$



贪心算法

讲授者 王爱娟

目录

贪心法的思;

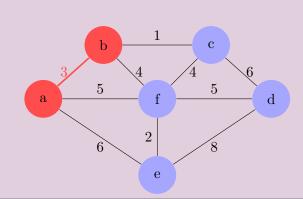
最小生成树 シPRIM算》

最小生成树 之Kruskal算

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续1)



贪心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算法

最小生成树 之Kruskal箅

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续1)

 $V - V_T = \{c,d,e,f\}$

 $V_T = \{a,b\}$ b 1 c d b d d

贪心芽泡

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算》

最小生成树 之Kruskal箅

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续1)

 $V - V_T = \{c,d,e,f\}$

 $V_T = \{a,b\}$ b 1 c d b d d

贪心芽汤

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算》

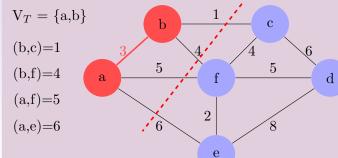
敢小生成树 之Kruskal算 ⅓

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续1)

 $V - V_T = \{c,d,e,f\}$



贪心芽》

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算》

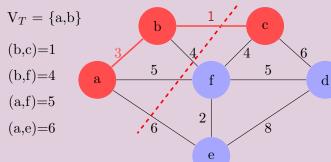
最小生成树 之Kruskal箅

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续1)

 $V - V_T = \{c,d,e,f\}$





贪心算法

讲授者 王爂娟

目录

贪心法的思想

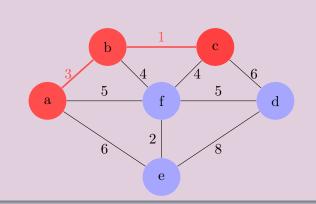
最小生成树 之PRIM算》

最小生成树 之Kruskal算 出

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续2)



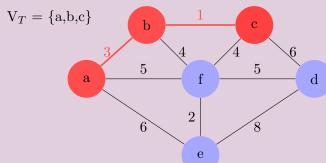
讲授者 王爱娟

Prim算法核心

总是找出权重最小的边(a,b), 其中a $\in V_T$, b $\in V - V_T$

Prim算法实例(续2)

 $V - V_T = \{d,e,f\}$



贪心芽泡

讲授者 王爱娟

目录

念心法的思

最小生成树 之PRIM算》

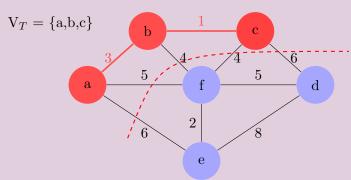
最小生成树 之Kruskal箅

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续2)

 $V - V_T = \{d,e,f\}$



贪心芽污

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算》

最小生成树 之Kruskal箅

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续2)

V - V $_T = \{d,e,f\}$

 $V_T = \{a,b,c\}$ (c,f)=4 (c,d)=6 (b,f)=4 (a,f)=5 (a,e)=6 (c,d)=6 (c,d)=6

贪心芽泡

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算治

最小生成树 之Kruskal箅

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续2)

V - V $_T = \{d,e,f\}$

 $V_T = \{a,b,c\}$ b 1 c (c,f)=4 (c,d)=6 (b,f)=4 (a,f)=5 (a,e)=6 e



贪心芽污

讲授者 王爱娟

目录

贪心法的思想

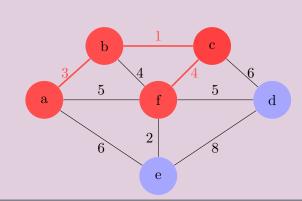
最小生成树 之PRIM算》

最小生成树 之Kruskal算 出

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续3)



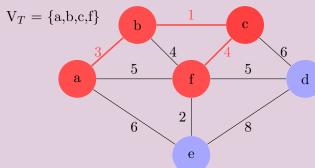
讲授者 王爱娟

Prim算法核心

总是找出权重最小的边(a,b), 其中a $\in V_T$, b $\in V - V_T$

Prim算法实例(续3)

 $V - V_T = \{d,e\}$



贪**心**算法 讲授者 王爱娟

Prim算法核心

总是找出权重最小的边(a,b), 其中a $\in V_T$, b $\in V - V_T$

Prim算法实例(续3)

 $V - V_T = \{d,e\}$

 $V_T = \{a,b,c,f\}$ b c d c d d d d

讲授者 王爱娟

Prim算法核心

总是找出权重最小的边(a,b), 其中a $\in V_T$, b $\in V - V_T$

Prim算法实例(续3)

 $V - V_T = \{d,e\}$

 $V_T = \{a,b,c,f\}$

b \mathbf{c} (c,d)=6

5 (f,d)=5a

(f,e)=2(a,e) = 68 6

e

d

贪心芽法 讲授者 王爱娟

Р

目录

最小生成树 之PRIM算治

最小生成树 之Kruskal箅

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续3)

 $V - V_T = \{d,e\}$

 $V_T = \{a,b,c,f\}$ b c (c,d)=6 (f,d)=5 (f,e)=2 (a,e)=6 b c d d



贪心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算》

最小生成树 之Kruskal箅 法

Prim算法核心

总是找出权重最小的边(a,b), 其中a $\in V_T$, b $\in V - V_T$

Prim算法实例(续4) c b 5 5 a d 2 6 e



贪心算法

讲授者 王爱娟

目录

贪心法的思?

最小生成树 之PRIM算》

最小生成树 之Kruskal算

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续4)

 $V - V_T = \{d\}$ $V_T = \{a,b,c,f,e\}$ c b 5 5 a d 2 6 e



贪心芽法

讲授者 王爱娟

目录

贪心法的思?

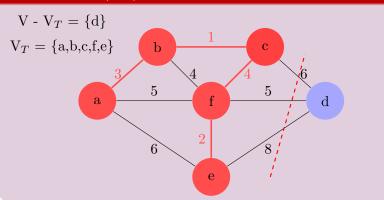
最小生成树 之PRIM算》

最小生成树 之Kruskal算

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续4)



贪心芽法

讲授者 王爱娟

目录

贪心法的思?

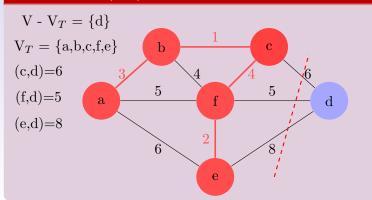
最小生成树 之PRIM算》

敢小生成树 之Kruskal算 ^壮

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续4)



贪心芽法

讲授者 王爱娟

目录

贪心法的思?

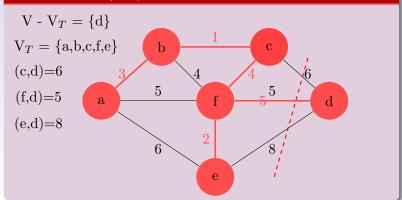
最小生成树 之PRIM算》

敢小生成树 之Kruskal算 ^汪

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续4)





贪心算法

讲授者 王爱娟

目录

含心法的思

最小生成树 之PRIM算治

最小生成树 之Kruskal算 注

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续5) b \mathbf{c} a d e



贪心芽法

讲授者 王爱娟

目录

贪心法的思

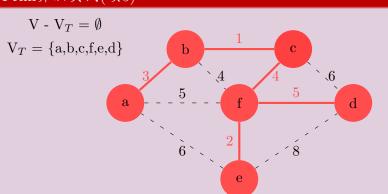
最小生成树 之PRIM算》

最小生成树 之Kruskal算

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续5)



贪心芽法

讲授者 王爱娟

目录

命心法的思?

最小生成树 之PRIM算》

最小生成树 之Kruskal算

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续5) $V - V_T = \emptyset$ $V_T = \{a, b, c, f, e, d\}$ b 最小生成树产生完成 a d e

贪心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算》

敢小生成树 之Kruskal算

Prim算法核心

总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

Prim算法实例(续5) $V - V_T = \emptyset$ $V_T = \{a, b, c, f, e, d\}$ b 最小生成树产生完成 a d w(T)=3+1+4+2+5=15e



贪心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算治

最小生成树 之Kruskal算

- 从图G中任意顶点vo开始产生最小生成树
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和V V_T
 - V_T 表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V V_T$ 。将 $b \to V_T$,并从 $V V_T$ 中删除
- 3 反复执行第2步,直至集合V-VT为空。



贪心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算法

敢小生成树 之Kruskal算

- 从图G中任意顶点v0开始产生最小生成树
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和 $V V_T$
 - V_T 表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V V_T$ 。将 $b \to V_T$,并 $d \to V_T$,并 $d \to V_T$,中删除
- ③ 反复执行第2步,直至集合V-V_T为空。



贪心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算法

最小生成树 之Kruskal算

- 从图G中任意顶点v0开始产生最小生成树
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和 $V V_T$
 - V_T表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V V_T$ 。将 $b \to V_T$,并 $b \to V_T$,中,并 $b \to V_T$,中删除
- ③ 反复执行第2步,直至集合V-V_T为空。



贪心算法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算治

最小生成树 之Kruskal箅 法

- 从图G中任意顶点v0开始产生最小生成树
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和V V_T
 - \bullet V_T 表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V V_T$ 。将b m入到 V_T 中,并从 $V V_T$ 中删除
- ③ 反复执行第2步,直至集合V-V√为空。



贪心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算治

最小生成树 之Kruskal算 法

- 从图G中任意顶点v0开始产生最小生成树
 - 顶点集V被划分为两个集合: $V_T = \{v_0\}$ 和 $V V_T$
 - \bullet V_T 表示生成树的顶点集
- ② 产生最小生成树:总是找出权重最小的边(a,b), 其中 $a\in V_T$, $b\in V-V_T$ 。将b加入到 V_T 中,并从 $V-V_T$ 中删除
- ③ 反复执行第2步,直至集合V-V_T为空。

Prim算法

贪心算法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

敢小生成树 之Kruskal算 法

```
Prim(G)
```

8: **end for** 9: return E_T

```
//构造最小生成树的Prim算法
//输入: 加权连通图G=<V,E>
//输出: 组成G的最小生成树的边集E_T
1: V_T = \{v_0\}
2: E_T = \emptyset
3: for i = 1 to |V| - 1 do
4: 找出权重最小的边(a,b),其中a\in V_T, b\in V - V_T
5: V_T = V_T \cup \{b\}
```

6: $V - V_T = V - V_T - \{b\}$ 7: $E_T = E_T \cup \{(a, b)\}$



贪心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 之PRIM算》

敢小生成树 之Kruskal算 法

Prim算法的时间效率分析

1: • • •

2: for i = 1 to |V| - 1 do

3: 找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

4: ...

- 算法时间复杂度主要由2-3行代码决定!
- FOR循环执行|V|-1次
- 确定算法的时间复杂度的核心是第3行代码
- 第3行代码的实现方式决定了它自身的复杂度



贪心算法

讲授者 王爱娟

目录

贪心法的思:

最小生成树 之PRIM算》

敢小生成树 之Kruskal算 法

Prim算法的时间效率分析

1: • • •

2: **for** i = 1 to |V| - 1 **do**

3: 找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

4: ...

- 算法时间复杂度主要由2-3行代码决定!
- FOR循环执行|V|-1次
- 确定算法的时间复杂度的核心是第3行代码
- 第3行代码的实现方式决定了它自身的复杂度



贪心算法

讲授者 王爱娟

目录

贪心法的思?

最小生成树 シPRIM算

最小生成树 之Kruskal葬 法

Prim算法的时间效率分析

1: • • •

2: for i = 1 to |V| - 1 do

3: 找出权重最小的边(a,b), 其中a $\in V_T$, b $\in V - V_T$

4: ...

- 算法时间复杂度主要由2-3行代码决定!
- FOR循环执行|V|-1次
- 确定算法的时间复杂度的核心是第3行代码
- 第3行代码的实现方式决定了它自身的复杂度



贪心算法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算》

最小生成树 之Kruskal∮ 法

Prim算法的时间效率分析

1: • • •

2: for i = 1 to |V| - 1 do

3: 找出权重最小的边(a,b), 其中 $a \in V_T$, $b \in V - V_T$

4: · · ·

- 算法时间复杂度主要由2-3行代码决定!
- FOR循环执行|V|-1次
- 确定算法的时间复杂度的核心是第3行代码
- 第3行代码的实现方式决定了它自身的复杂度



Prim算法效率分析 - 无序数组实现优先队列

贪心芽法

讲授者 王愛娟

目录

贪心法的思想

最小生成树 之PRIM算治

最小生成树 之Kruskal算 法

1.无序数组实现优先队列

图是由权重矩阵来表示,而优先队列由一个无序数组来实现



Prim算法效率分析 - 无序数组实现优先队列

贪心算法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算記

最小生成树 之Kruskal∮ 法

1.无序数组实现优先队列

图是由权重矩阵来表示, 而优先队列由一个无序数组来实现

复杂度分析

- 第三行代码用数组来实现,每次从数组中选出最小的权值,复杂度为O(|V|);
- 由于for循环,一共需要向 V_T 中加入|V|-1个节点,总的复杂度为 $O(|V|^2)$;

2.最小堆实现优先队列

贪**心**算法 讲授者 王舜娟

目录 贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal亸 法

时间效率分析

- 最坏情况下, 每一层的节点的键都会交换到最底层;
- 移动一次需要两次比较:是否执行交换和找出最大的子女;
- 第i层的每个节点(一共有 2^i 个)需要2(h-i)次键值比较;
- $C_{worst}(n) = \sum_{i=0}^{log_2 n 1} 2^i * 2(log_2 n i) = 2(n log_2(n+1))$

堆的操作

插入操作:与堆构造堆相似,最坏情况下的复杂度为 $O(log_2n)$ (树的高度)删除操作:

- 要删除的节点和堆的最后一个节点交换, 然后删除;
- "堆化" (自底向上构造堆步骤2-3) 删除后的树;



贪**心**算法 讲授者 王爱娟

目录

贪心法的思想 P1444

取小生成柄 之PRIM算法

最小生成树 之Kruskal算 注

基于最小堆的Prim算法的时间效率分析

使用最小堆存储 $V - V_T$, 其中键值计算如下:

- 如果V V_T 中的顶点与 V_T 相邻,则顶点键值为该顶点离 V_T 的最短边的权重
- 如果V V_T 中的顶点与 V_T 不相邻,则顶点键值为 ∞
- 堆的根即为当前 $V V_T$ 离 V_T 最近的顶点!



基于最小堆的Prim算法的时间效率分析

使用最小堆存储 $V - V_T$, 其中键值计算如下:

- 如果V V_T 中的顶点与 V_T 相邻,则顶点键值为该顶点离 V_T 的最短边的权重
- 如果V V_T 中的顶点与 V_T 不相邻,则顶点键值为 ∞
- 堆的根即为当前 $V V_T$ 离 V_T 最近的顶点!

贪**心**算法 讲授者 王愛娟

口水 贪心法的思想 最小生成树

之PRIM算法 最小生成树



基于最小堆的Prim算法的时间效率分析

使用最小堆存储 $V - V_T$, 其中键值计算如下:

- 如果V V_T 中的顶点与 V_T 相邻,则顶点键值为该顶点离 V_T 的最短边的权重
- 如果V V_T 中的顶点与 V_T 不相邻,则顶点键值为 ∞
- 堆的根即为当前 $V V_T$ 离 V_T 最近的顶点!

|录 |心法的思想

贪心算法

讲授者 王受娟

贪心法的思想 最小生成树 之PRIM算法



基于最小堆的Prim算法的时间效率分析

使用最小堆存储 $V - V_T$, 其中键值计算如下:

- •如果 $V V_T$ 中的顶点与 V_T 相邻,则顶点键值为该顶点离 V_T 的最短边的权重
 - 如果 $V V_T$ 中的顶点与 V_T 不相邻,则顶点键值为 ∞

找出权重最小的边(a,b), 其中a $\in V_T$, b $\in V - V_T$ 的实现如

• 堆的根即为当前 $V - V_T$ 离 V_T 最近的顶点!

下:

- 1: $u = EXTRACT-HEAP(V V_T)$
- 2: for v is adjacent to u do
- 3: if $v \in V V_T$ and w(u,v) < key[v] then
- 4: $\operatorname{key}[v] = \operatorname{w}(u,v)$
- 5: end if

6: end for

目录 贪心法的思考

贪心算法

讲授者 王受娟

最小生成树 之PRIM算法

最小生成树 之Kruskal算 法

Prim算法的时间开销分析(续2)

贪心算法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

取小生成树 之Kruskal葬 法

```
for i = 1 to |V| - 1 do
找出权重最小的边(a,b),其中a \in V_T,b \in V - V_T
end for
```

上面的代码可以等价转换如下:

```
1: while V - V_T \neq \emptyset do

2: u = \text{EXTRACT-HEAP}(V - V_T)

3: for v is adjacent to u do

4: if v \in V - V_T and w(u,v) < \text{key}[v] then

5: \text{key}[v] = w(u,v)

6: end if

7: end for

8: end while

实现第2行代码的复杂度为O(\log |V|)
```



Prim算法的时间开销分析(续3)

```
贪心芽法
讲授者 王爱娟
```

目录 含心法的思想

最小生成树 シPRIM 質法

之PRIM算法 最小生成树

最小生成树 之Kruskal算 法

```
1: while V - V_T \neq \emptyset do
2: u = EXTRACT-HEAP(V - V_T)
```

3: **for** v is adjacent to u **do**

4: if $v \in V - V_T$ and w(u,v) < key[v] then

5: $\operatorname{key}[v] = \operatorname{w}(u,v)$

6: end if

7: end for 8: end while



Prim算法的时间开销分析(续3)

- 贪心算法 讲授者 王爱娟

- 1: while V $V_T \neq \emptyset$ do $u = EXTRACT-HEAP(V - V_T)$ 2:
- for v is adjacent to u do 3:
 - if $v \in V V_T$ and w(u,v) < key[v] then key[v] = w(u,v)
 - end if
 - end for
- 8: end while

基于最小堆的Prim算法的时间效率分析

4: 5:

6:

7:

- 第2行代码的时间复杂度为O(log |V|)
- 1-2行代码的时间复杂度为(|V| − 1)O(log |V|) • 1,3行代码(两次循环)总共执行多少次? 2|E|次
- 每一次FOR循环, key[v] = w(u,v)可能被执行, 这会触 发堆的重构, 其时间复杂度为 $O(\log |V|)$

算法时间复杂度为(|V|-1+|E|) O(log |V|))



近心算法

讲授者 王爱娟

目录

贪心法的思

最小生成树 >DDIMG

之PRIM算》

之Kruskal算

最小生成树

加权连通图G=<V,E>的一棵最小生成树是图的一棵权重最小的生成树



贪心芽法

讲授者 王爱娟

目录

贪心法的思?

最小生成树 之PRIM算?

最小生成树 之Kruskal算

最小生成树

加权连通图G=<V,E>的一棵<mark>最小生成树</mark>是图的一棵权重最小的生成树

Kruskal算法的思想

总是选取E中最短的边来构成树。是否能够产生最小生成树,试一试!

- 图中边集E按照权重的非递减顺序进行排序
- 总是选取边值最小的边构成树
- 所选的边不能与已产生的树构成回路!



贪心算法 最小生成树

加权连通图G=<V.E>的一棵最小生成树是图的一棵权重最 小的生成树

Kruskal算法的思想

总是选取E中最短的边来构成树。是否能够产生最小生成

- 树. 试一试! 图中边集E按照权重的非递减顺序进行排序
 - 总是选取边值最小的边构成树
 - 所选的边不能与已产生的树构成回路!
- 终止条件
 - 为|V|-1

图的最小生成树必须包含所有顶点V. 它的边数只能

终止条件:选择|V|-1条边后终止。最小生成树生成!

讲授者 王受娟



最小生成树

加权连通图G=<V,E>的一棵最小生成树是图的一棵权重最

小的生成树

Kruskal算法的思想

总是选取E中最短的边来构成树。是否能够产生最小生成树, 试一试!

- 图中边集E按照权重的非递减顺序进行排序
- 总是选取边值最小的边构成树
- 所选的边不能与已产生的树构成回路!

终止条件

- 业余什 ● 图的最小生成树必须包含所有顶点V. 它的边数只能
- 为|V|-1 • 终止条件:选择|V|-1条边后终止,最小生成树生成!

讲授者 王夢娟

目录

最小生成树 シPRIM算法

最小生成树

之Kruskal算 法

贪心算法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal算

```
Kruskal(G)
```

```
//构造最小生成树的Kruskal算法 //输入: 加权连通图G=<V,E> //输出: E_T
```

1: 按照边的权重非递减顺序对集合E排序

```
2: E_T = \emptyset; encounter = 0;
```

3: **while** encounter < |V|-1 **do**

4: **if** $E_T \cup (a,b)$ 无回路 **then**

5: $E_T = E_T \cup (a,b);$

6: encounter = encounter + 1;

7: end if

8: end while

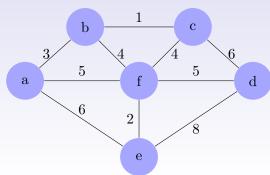
9: return E_T



贪心算法

讲授者 王受娟

- 一个图 $G=\langle V,E\rangle$ 如下所示, $V=\{a,b,c,d,e,f\}$, $E=\{(a,b,3),(a,f,5),$ (a,e,6),(b,c,1),(b,f,4),(c,d,6),(c,f,4),(d,e,8),(d,f,5),(f,e,2)
 - 第一步对E进行排序: $E=\{(b,c,1),(f,e,2),(a,b,3),(b,f,4),$ (c,f,4),(a,f,5),(d,f,5),(c,d,6),(a,e,6),(d,e,8)
 - $E_T = \emptyset$ (图中红线组成), encounter = 0;
 - 依次选择最小边构成生成树, 但所选边不能构成回路



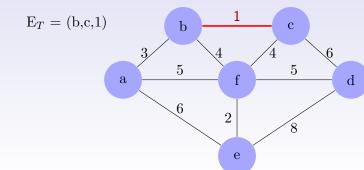


贪心算法

讲授者 王受娟

一个图 $G=\langle V,E\rangle$ 如下所示, $V=\{a,b,c,d,e,f\}$, $E=\{(a,b,3),(a,f,5),$ (a,e,6),(b,c,1),(b,f,4),(c,d,6),(c,f,4),(d,e,8),(d,f,5),(f,e,2)

- 第一步对E进行排序: $E=\{(b,c,1),(f,e,2),(a,b,3),(b,f,4),$ (c,f,4),(a,f,5),(d,f,5),(c,d,6),(a,e,6),(d,e,8)
- $E_T = \emptyset$ (图中红线组成), encounter = 0;
- 依次选择最小边构成生成树, 但所选边不能构成回路





贪心算法

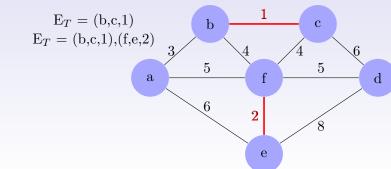
讲授者 王爱娟

贪心法的思素

最小生成树 之PRIM算法

取小王成何 之Kruskal算 法

- 一个图G= $\langle V, E \rangle$ 如下所示, $V = \{a,b,c,d,e,f\}$, $E = \{(a,b,3),(a,f,5),(a,e,6),(b,c,1),(b,f,4),(c,d,6),(c,f,4),(d,e,8),(d,f,5),(f,e,2)\}$
 - 第一步对E进行排序: E={(b,c,1),(f,e,2),(a,b,3),(b,f,4), (c,f,4),(a,f,5),(d,f,5),(c,d,6),(a,e,6),(d,e,8)}
 - $E_T = \emptyset$ (图中红线组成), encounter = 0;
 - 依次选择最小边构成生成树, 但所选边不能构成回路





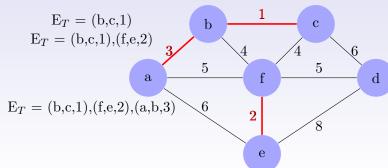
贪**心**算法 讲授者 王爱娟

目录 贪心法的思考

最小生成树 之PRIM算法

最小生成树 之Kruskal算 法 一个图G= $\langle V, E \rangle$ 如下所示, $V = \{a,b,c,d,e,f\}$, $E = \{(a,b,3),(a,f,5)$, $(a,e,6),(b,c,1),(b,f,4),(c,d,6),(c,f,4),(d,e,8),(d,f,5),(f,e,2)\}$ • 第一步对E进行排序: $E = \{(b,c,1),(f,e,2),(a,b,3),(b,f,4)$

- (c,f,4),(a,f,5),(d,f,5),(c,d,6),(a,e,6),(d,e,8)
- $E_T = \emptyset$ (图中红线组成), encounter = 0;
- 依次选择最小边构成生成树,但所选边不能构成回路





贪心芽法

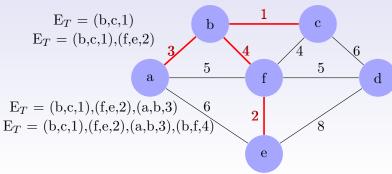
讲授者 王爱娟

贪心法的思想

最小生成树 之PRIM算法

取小生成树 之Kruskal算 去

- 一个图G= $\langle V,E \rangle$ 如下所示, $V=\{a,b,c,d,e,f\}$, $E=\{(a,b,3),(a,f,5),(a,e,6),(b,c,1),(b,f,4),(c,d,6),(c,f,4),(d,e,8),(d,f,5),(f,e,2)\}$
 - 第一步对E进行排序: E={(b,c,1),(f,e,2),(a,b,3),(b,f,4),(c,f,4),(a,f,5),(d,f,5),(c,d,6),(a,e,6),(d,e,8)}
 - $E_T = \emptyset$ (图中红线组成), encounter = 0;
 - 依次选择最小边构成生成树, 但所选边不能构成回路





贪心芽法

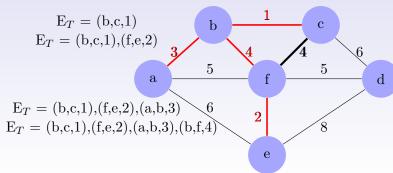
讲授者 王爱娟

贪心法的思想

最小生成树 之PRIM算法

取小王成衲 之Kruskal算 法

- 一个图G= $\langle V,E \rangle$ 如下所示, $V=\{a,b,c,d,e,f\}$, $E=\{(a,b,3),(a,f,5),(a,e,6),(b,c,1),(b,f,4),(c,d,6),(c,f,4),(d,e,8),(d,f,5),(f,e,2)\}$
 - 第一步对E进行排序: E={(b,c,1),(f,e,2),(a,b,3),(b,f,4),(c,f,4),(a,f,5),(d,f,5),(c,d,6),(a,e,6),(d,e,8)}
 - $E_T = \emptyset$ (图中红线组成), encounter = 0;
 - 依次选择最小边构成生成树, 但所选边不能构成回路





贪心算法

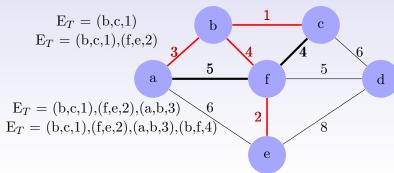
讲授者 王爱娟

or 不 贪心法的思想

最小生成树 之PRIM算法

最小生成树 之Kruskal算 ^注 一个图G= $\langle V,E \rangle$ 如下所示, $V=\{a,b,c,d,e,f\}$, $E=\{(a,b,3),(a,f,5),(a,e,6),(b,c,1),(b,f,4),(c,d,6),(c,f,4),(d,e,8),(d,f,5),(f,e,2)\}$

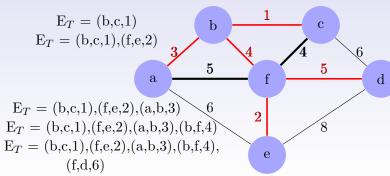
- 第一步对E进行排序: E={(b,c,1),(f,e,2),(a,b,3),(b,f,4),(c,f,4),(a,f,5),(c,d,6),(a,e,6),(d,e,8)}
- $E_T = \emptyset($ 图中红线组成), encounter = 0;
- 依次选择最小边构成生成树, 但所选边不能构成回路





一个图 $G=\langle V,E\rangle$ 如下所示, $V=\{a,b,c,d,e,f\}$, $E=\{(a,b,3),(a,f,5),(a,e,6),(b,c,1),(b,f,4),(c,d,6),(c,f,4),(d,e,8),(d,f,5),(f,e,2)\}$ • 第一步对E进行排序: $E=\{(b,c,1),(f,e,2),(a,b,3),(b,f,4),($

- (c,f,4),(a,f,5),(d,f,5),(c,d,6),(a,e,6),(d,e,8)
- $E_T = \emptyset$ (图中红线组成), encounter = 0;
- 依次选择最小边构成生成树,但所选边不能构成回路



Kruskal算法实例(续)

负心算法

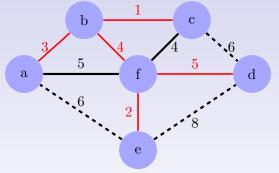
讲授者 王愛娟

目录

贪心法的思

最小生成树 シPRIM算:

最小生成树 之Kruskal 最小生成树的权值为:w(T)=1+2+3+4+5=15



Kruskal算法的关键

如何确保新加入的边与已产生的树不构成回路?

判断新加入边是否构成回路

讲授者 王爱娟

贪心算法

日 不 会 小 注 码 田

最小生成树 クPRIM算法

最小生成树

之Kruskal算 法 $\mathbf{A} = \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}, \mathbf{f}\} \xrightarrow{partition} \{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{f\}\}$

union操作和find操作

- union(a,b) = $\{a,b\}$
- union(a,c) = $\{a,b,c\}$
- $\bullet \ union(d,e) = \{d,e\}$
- $union(a,d) = \{a,b,c,d,e\}$
- 两个元素的union = 包含 该元素的集合的并



判断新加入边是否构成回路

贪心草法

讲授者 王愛娟

1录

贪心法的思想

最小生成树 之PRIM算》

最小生成树 之Kruskal算 $A = \{a,b,c,d,e,f\} \xrightarrow{partition} \{a\},\{b\},\{c\},\{d\},\{e\},\{f\}\}$

union操作和find操作

- $union(a,b) = \{a,b\}$
 - $\bullet \ union(a,c) = \{a,b,c\}$
 - $union(d,e) = \{d,e\}$
 - $\bullet \ union(a,d) = \{a,b,c,d,e\}$
 - 两个元素的union = 包含 该元素的集合的并

find操作

- $\Rightarrow find(a) = find(b) = \{a,b\}$
- find(a)=find(b)=find(c) ={a,b,c}
- $find(d)=find(e)=\{d,e\}$
- find(a) = find(d)= $\{a,b,c,d,e\}$
- find一个元素= 返回包含 该元素的集合



判断新加入边是否构成回路

贪心算法

讲授者 王爱娟

贪心法的思考

最小生成树 とPRIM算る

最小生成树 之Kruskal算 壮 A = {a,b,c,d,e,f} $\xrightarrow{partition}$ {a},{b},{c},{d},{e},{f}

union操作和find操作

- $union(a,b) = \{a,b\}$
 - $\bullet \ union(a,c) = \{a,b,c\}$
 - $union(d,e) = \{d,e\}$
 - $\bullet \ union(a,d) = \{a,b,c,d,e\}$
 - 两个元素的union = 包含 该元素的集合的并

find操作

- $\Rightarrow find(a) = find(b) = \{a, b\}$
- find(a)=find(b)=find(c) ={a,b,c}
- $find(d) = find(e) = \{d,e\}$
- find(a)=find(d)= $\{a,b,c,d,e\}$
- find一个元素= 返回包含 该元素的集合

union和find操作能否用于判断回路?



6/2.算法

讲授者 王爱娟

]录

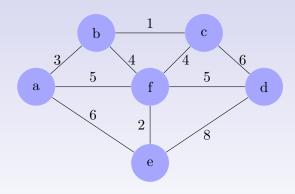
贪心法的思?

最小生成树

之PRIM算法

最小生成树

之Kruskal算 法 $\begin{aligned} \mathbf{V} &= \{ \mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}, \mathbf{f} \} \xrightarrow{part} \{ a \}, \{ b \}, \{ c \}, \{ d \}, \{ e \}, \{ f \} \\ \mathbf{E}_T &= \{ \emptyset \}, \ \mathbf{V}_T &= \{ \emptyset \} \end{aligned}$



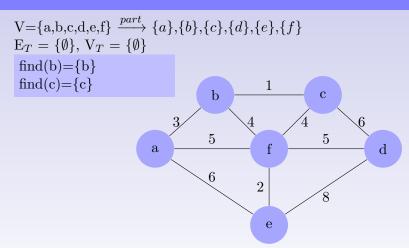


```
讲授者 王爱娟
```

最小生成树 之PRIM算法

最小生成树 之Kruskal算

之Kruskal算 法





```
贪心禁法
讲授者 王要娟
目录

贪心法的思想
```

最小生成树 之Kruskal算 $V = \{a,b,c,d,e,f\} \xrightarrow{part} \{a\},\{b\},\{c\},\{d\},\{e\},\{f\}\}$ $E_T = \{\emptyset\}, V_T = \{\emptyset\}$ $find(b) = \{b\}$ $find(c) = \{c\}$ b union(b,c)= $\{b,c\}$ 5 5 a d 2 8 e



```
V = \{a,b,c,d,e,f\} \xrightarrow{part} \{a\},\{b\},\{c\},\{d\},\{e\},\{f\}\}
  贪心算法
                E_T = \{\emptyset\}, V_T = \{\emptyset\}
讲授者 王爱娟
                 find(b) = \{b\}
                 find(c) = \{c\}
                                                             b
                 union(b,c)=\{b,c\}
                                                             5
                                                                                      5
                                                a
                                                             6
                                                                        2
                 find(f) = \{f\} find(e) = \{e\}
                                                                                       8
                                                                          e
```

d



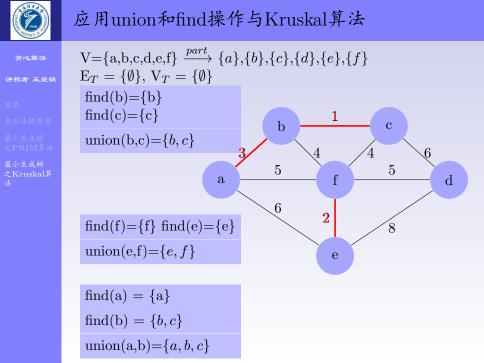
 $V = \{a,b,c,d,e,f\} \xrightarrow{part} \{a\},\{b\},\{c\},\{d\},\{e\},\{f\}\}$ 贪心算法 $E_T = \{\emptyset\}, V_T = \{\emptyset\}$ 讲授者 王爱娟 $find(b) = \{b\}$ $find(c) = \{c\}$ b union(b,c)= $\{b,c\}$ 5 5 a d 2 $find(f) = \{f\} find(e) = \{e\}$ 8 union(e,f)= $\{e, f\}$ e



 $V = \{a,b,c,d,e,f\} \xrightarrow{part} \{a\},\{b\},\{c\},\{d\},\{e\},\{f\}\}$ 贪心算法 $E_T = \{\emptyset\}, V_T = \{\emptyset\}$ 讲授者 王爱娟 $find(b) = \{b\}$ $find(c) = \{c\}$ b union(b,c)= $\{b,c\}$ 5 a 6 2 $find(f) = \{f\} find(e) = \{e\}$ union(e,f)= $\{e, f\}$ $find(a) = \{a\}$ $find(b) = \{b, c\}$

5 d 8

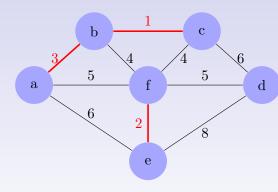
e





讲授者 王爱娟

 ${a,b,c},{d},{e,f}$ $E_T = \{(a, b), (b, c), (e, f)\},\$





```
比拟李 工 巫 42
```

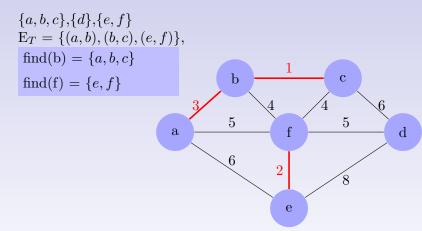
讲授者 王爱娟

日水 贪心法的思想

最小生成树 之PRIM算法

最小生成树

之Kruskal算 法



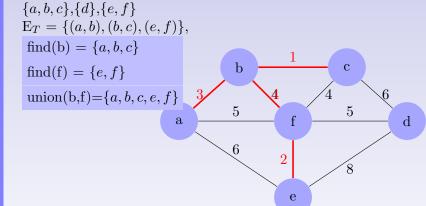


```
贪心芽法
讲授者 王爱娟
```

贪心法的思想 最小生成树

最小生成树 之PRIM算法 最小生成树

之Kruskal算 法



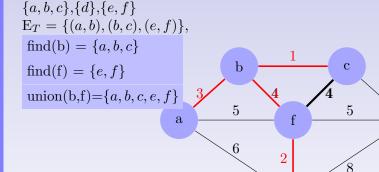


```
贪心芽法
讲授者 王爱娟
目录
```

贪心法的思想 最小生成树

最小生成树 之PRIM算法 最<mark>小生成树</mark>

之Kruskal算 法



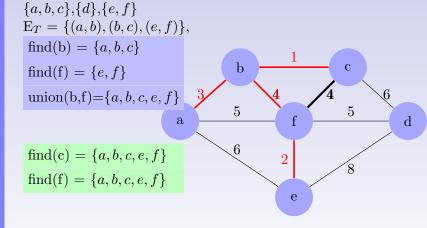
e



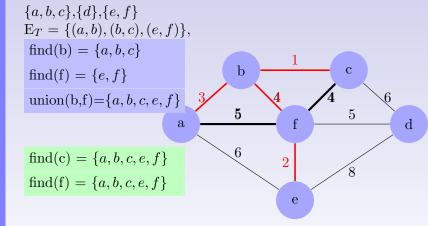
```
贪心算法
讲授者 王荣娟
目录
贪心法的思想
最小生成树
之PRIM算法
```

之PRIM算法 最小生成树 えKruskal算

取小王成衲 之Kruskal算 法







贪心芽法讲授者 王荣娟 目录 贪心法的思想 最小生成树 之PRIM算法

```
\{a,b,c\},\{d\},\{e,f\}
E_T = \{(a,b), (b,c), (e,f)\},\
 find(b) = \{a, b, c\}
                                          b
 find(f) = \{e, f\}
 union(b,f)=\{a, b, c, e, f\}
                                         5
                                                               5
                              a
 find(c) = \{a, b, c, e, f\}
                                                  2
                                                               8
 find(f) = \{a, b, c, e, f\}
                                                    e
 find(a) = \{a, b, c, e, f\}
 find(f) = \{a, b, c, e, f\}
```



贪心算法 讲授者 王爱娟

 $find(f) = \{e, f\}$

 $find(b) = \{a, b, c\}$

 $\{a,b,c\},\{d\},\{e,f\}$

 $find(c) = \{a, b, c, e, f\}$

 $find(f) = \{a, b, c, e, f\}$

 $find(a) = \{a, b, c, e, f\}$

 $find(f) = \{a, b, c, e, f\}$

 $E_T = \{(a,b), (b,c), (e,f)\},\$

union(b,f)= $\{a,b,c,e,f\}$

a

应用union和find操作与Kruskal算法(续1)

b

5

2

 $find(f) = \{a, b, c, e, f\}$

 $find(d) = \{d\}$

e

union(f,d)= $\{a, b, c, d, e, f\}$

5

8



 $\{a,b,c\},\{d\},\{e,f\}$

 $find(b) = \{a, b, c\}$

 $find(f) = \{e, f\}$

 $E_T = \{(a,b), (b,c), (e,f)\},\$

union(b,f)= $\{a,b,c,e,f\}$

 $find(c) = \{a, b, c, e, f\}$

 $find(f) = \{a, b, c, e, f\}$

 $find(a) = \{a, b, c, e, f\}$

 $find(f) = \{a, b, c, e, f\}$

b

5

2

 $find(f) = \{a, b, c, e, f\}$

 $find(d) = \{d\}$

e

union(f,d)= $\{a, b, c, d, e, f\}$

a

5

8

贪心算法 讲授者 王爱娟





贪心芽法

讲授者 王爱娟

目录

贪心法的思:

最小生成树

ZPRIM井 »

之Kruskal算

应用union和find判断回路

若边(a,b)被加入到生成树。如果 $\mathrm{find}(a)=\mathrm{find}(b)$ 则表明该边会导致回路,不能添加;否则,则可以添加。



贪心算法

讲授者 王爱娟

应用union和find操作与Kruskal算法(续2)

应用union和find判断回路

若边(a,b)被加入到生成树。如果find(a)=find(b)则表明该边 会导致回路, 不能添加; 否则, 则可以添加。

Kruskal(G)

//构造最小生成树的Kruskal算法 //输入: 加权连通图G=<V,E>

//输出: E_T 1: 按照边的权重非递减顺序对集合E排序

2: $E_T = \emptyset$; encounter = 0; 3: while encounter < |V|-1 do

取下一条最短边(a,b) 5: **if** find(a) \neq find(b) **then**

 $E_T = E_T \cup (a,b)$; union(a,b); encounter++;

end if

7: 8: end while 9: return E_T

6:



贪心算法

讲授者 王爱娟

目录

贪心法的思想

最小生成树 之PRIM算法

最小生成树

之Kruskal算 法 Kruskal(G)

1: 按照边的权重非递减顺序对集合E排序

2: **while** encounter < |V|-1 **do**

3: **if** $find(a) \neq find(b)$ **then**

 $4: \quad union(a,b);$

5: end if

6: end while

7: return \mathbf{E}_T



贪心算法

讲授者 王爱娟

目录

是小人名科

之PRIM算法

之Kruskal算 去

Kruskal(G)

1: 按照边的权重非递减顺序对集合E排序

2: while encounter < |V|-1 do

3: **if** $find(a) \neq find(b)$ **then**

4: union(a,b);

5: end if

6: end while

7: return \mathbf{E}_T

- 第1行代码的时间复杂度为 $O(|E|\log|E|)$ (快速排序)
- 一次union操作的时间复杂度为O(1)
- 2-3行代码的m次find操作的时间复杂度为 $O(m \log |V|)$
- 总的时间复杂度为 $O(|V|)+O(|E|\log|E|)+O(m\log|V|)$



贪心算法

讲授者 王爱娟

目录

最小生成树

之PRIM算》 器小生成耕

之Kruskal算 去

Kruskal(G)

1: 按照边的权重非递减顺序对集合E排序

2: **while** encounter < |V|-1 **do**

3: **if** $find(a) \neq find(b)$ **then**

4: union(a,b);

5: end if

6: end while

7: return \mathbf{E}_T

- 第1行代码的时间复杂度为O(|E|log|E|)(快速排序)
- 一次union操作的时间复杂度为O(1)
- 2-3行代码的m次find操作的时间复杂度为 $O(m \log |V|)$
- 总的时间复杂度为 $O(|V|)+O(|E|\log |E|)+O(m\log |V|)$



贪心算法

讲授者 王爱娟

目录

贪心法的思想

取小生成树 之PRIM算法

之Kruskal算 法

Kruskal(G)

1: 按照边的权重非递减顺序对集合E排序

2: while encounter < |V|-1 do

3: **if** $find(a) \neq find(b)$ **then**

 $4: \quad \text{union}(a,b);$

5: end if

6: end while

7: return \mathbf{E}_T

- 第1行代码的时间复杂度为O(|E|log|E|)(快速排序)
- 一次union操作的时间复杂度为O(1)
- 2-3行代码的m次find操作的时间复杂度为 $O(m \log |V|)$
- 总的时间复杂度为 $O(|V|)+O(|E|\log|E|)+O(m\log|V|)$

贪心草法

讲授者 王受娟

目录 今小:24.44.田村

最小生成树 之PRIM算法

Kい王成州 とKruskal算 去

Kruskal(G)

- 1: 按照边的权重非递减顺序对集合E排序
- 2: while encounter < |V|-1 do 3: if find(a) \neq find(b) then
- 4: union(a,b);
- 5: end if
- 6: end while
- 7: return \mathbf{E}_T

- 第1行代码的时间复杂度为O(|E| log |E|)(快速排序)
- 一次union操作的时间复杂度为O(1)
- 2-3行代码的m次find操作的时间复杂度为 $O(m \log |V|)$
- 总的时间复杂度为 $O(|V|)+O(|E|\log |E|)+O(m\log |V|)$



Prim算法与Kruskal算法的比较

贪心算法

讲授者 王爱娟

1录

贪心法的思想

最小生成树 之PRIM算法

敢小生成树 之Kruskal∮ 注

Prim算法与Kruskal算法的比较

- 贪心法的思想: 总是选择当前最优的解
- ●不同的贪心选择,导致不同的贪心算法。因此,贪心算法的好坏关键在于正确的贪心选择。 Jump to different greedy choices
- Prim的贪心选择: 距离最近的点
- Kruskal的贪心选择: 权值最短的边