

信奥算法

图论要素

现场挑战

943

用纸和笔概括题目大意
已知什么求什么

限时3分钟

图论要素识别

无向图

正权图

多源单汇

单源多汇

可能不连通

$n \leq 20000$, $m \leq 200000$, 稀疏图

算法选择

Dijkstra堆优化

$O(E \log V)$

Dijkstra无优化

```
fill(dst, dst+n+9, INF);
dst[1]=0;
for(int k=1; k<=n-1; k++){
    int u=n+1;
    for(int j=1; j<=n; j++)
        if(!ok[j] && dst[j]<dst[u]) u=j;
    if(dst[u]>=INF) break;
    ok[u]=1;
    for(int i=0; i<to[u].size(); i++)
        dst[to[u][i]]=min(dst[to[u][i]], dst[u]+w[u][i]);
}
```

Dijkstra堆优化

```
6  bool ok[N];
7  ll n,m,dst[N],x[N];
8  vector<ll> to[N],w[N];
9  struct Node{
10     ll c,u;
11     bool operator<(const Node&a)const{
12         return c>a.c;
13     }
14 };

17  priority_queue<Node> q;
```

Dijkstra堆优化

```
15 void Dijkstra(){
16     fill(dst,dst+n+9,INF);
17     priority_queue<Node> q;
18     q.push((Node){0,1});
19     dst[1]=0;
20     while(!q.empty()){
21         ll u=q.top().u; q.pop();
22         if(ok[u])continue;
23         ok[u]=1;
24         for(ll i=0;i<to[u].size();i++){
25             ll v=to[u][i];
26             if(!ok[v]&&dst[v]>dst[u]+w[u][i]){
27                 dst[v]=dst[u]+w[u][i];
28                 q.push((Node){dst[v],v});
29             }
30         }
31     }
32 }
```

单源最短路 SSSP

非负权图	Dijkstra	$O(V^2)$
	Dijkstra堆优化	$O(E \log V)$
负权图	Bellman-Ford	$O(EV)$
	SPFA	$O(EV)$
		大部分情况 实际很快

Bellman-Ford分析+改进

```
10 void BellmanFord(){
11     fill(d,d+n+1,INF);
12     d[1]=0;
13     for(ll k=1;k<=n-1;k++)
14         for(ll i=1;i<=n;i++)
15             for(ll j=0;j<to[i].size();j++)
16                 d[to[i][j]]=min(d[to[i][j]],d[i]+w[i][j]);
17 }
```

共 $n-1$ 轮迭代更新

若某轮迭代所有点都没更新

若某点在某轮迭代没更新

该轮更新过的点入队

SPFA

shortest path
faster algorithm

in[i]是什么

```
9 void spfa(){
10     fill(dst,dst+n+9,INF);
11     queue<ll> q;
12     dst[1]=0; in[1]=1; q.push(1);
13     while(!q.empty()){
14         ll v,u=q.front(); q.pop(); in[u]=0;
15         for(ll i=0;i<to[u].size();i++)
16             if(dst[v=to[u][i]]>dst[u]+w[u][i]){
17                 dst[v]=dst[u]+w[u][i];
18                 if(!in[v]){
19                     q.push(v);
20                     in[v]=1;
21                 }
22             }
23     }
24 }
```

SPFA

shortest path
faster algorithm

```
8 void spfa(){
9     fill(dst,dst+n+9,INF);
10    queue<int> q;
11    dst[1]=0; in[1]=1; q.push(1);
12    while(!q.empty()){
13        int v,u=q.front(); q.pop(); in[u]=0;
14        for(int i=0;i<to[u].size();i++)
15            if(dst[v=to[u][i]]>dst[u]+w[u][i]){
16                dst[v]=dst[u]+w[u][i];
17                if(!in[v])q.push(v);
18            }
19    }
20 }
```

in[v]忘了改
in[v]=1;
会WA吗?

卡SPFA

请构造一类正权图

放Dijkstra堆优化AC

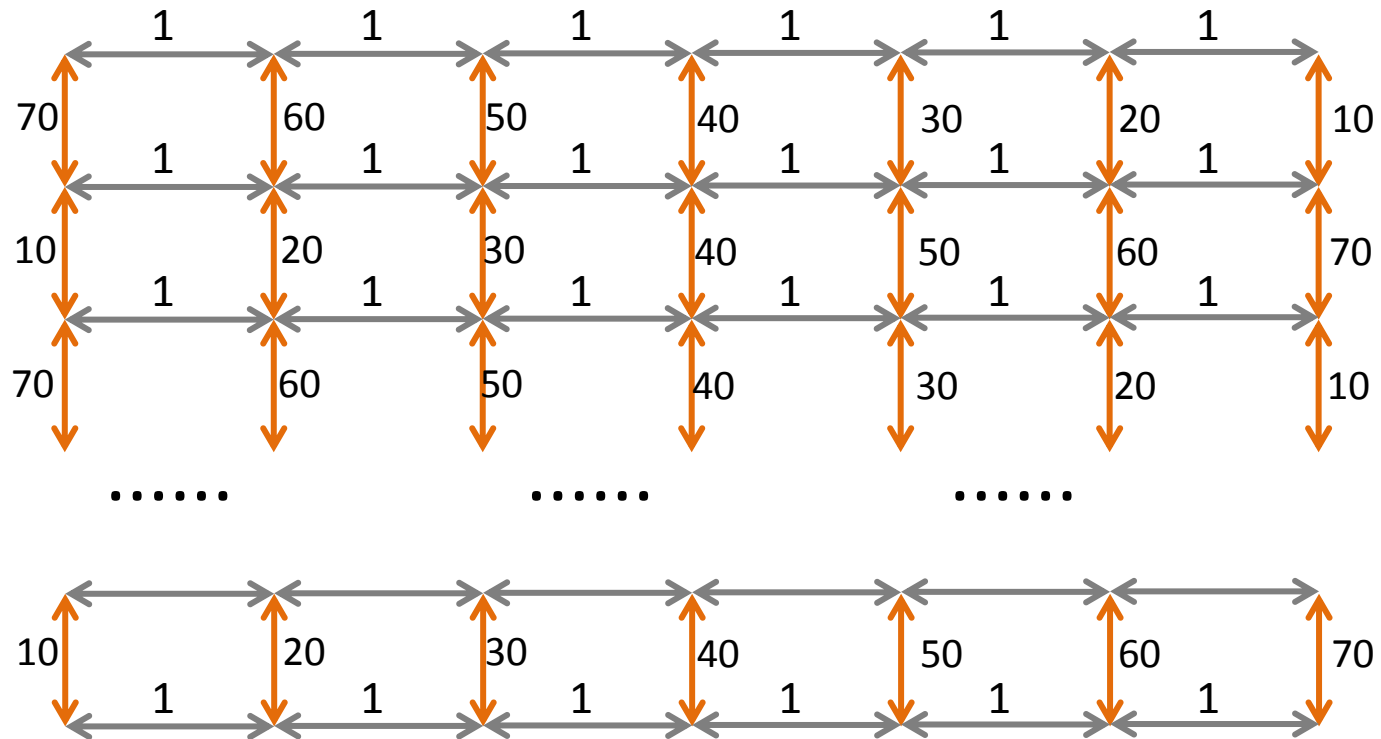
卡SPFA超时

达到最差情况复杂度
 $O(VE)$

3分钟后请同学分享

卡SPFA 网格图

SPFA退化到 $O(VE)$,
需要每一轮都有 $O(E)$ 边更新
更新轮次 $O(V)$ 轮



判负环

判负环

有向图

负权图

判断负权回路

有可能不连通

算法选择

Bellman-Ford

SPFA

SPFA判负环错误代码

```
8 int spfa(){
9     queue<int> q;
10    q.push(1);
11    while(!q.empty()){
12        int v,u=q.front(); q.pop(); in[u]=0;
13        for(int i=0;i<to[u].size();i++)
14            if(d[v=to[u][i]]>d[u]+w[u][i]){
15                d[v]=d[u]+w[u][i];
16                if(!in[v])in[v]=1,cnt[v]++,q.push(v);
17                if(cnt[v]>n)return 1;
18            }
19    }
20    return 0;
21 }
```

错在哪?

和1号点
不连通的点
都没有入队
没有资格更新

SPFA判负环正确代码

```
8 int spfa(){
9     queue<int> q;
10    for(int i=1;i<=n;i++)q.push(i),in[i]=cnt[i]=1;
11    while(!q.empty()){
12        int v,u=q.front(); q.pop(); in[u]=0;
13        for(int i=0;i<to[u].size();i++)
14            if(d[v=to[u][i]]>d[u]+w[u][i]){
15                d[v]=d[u]+w[u][i];
16                if(!in[v])in[v]=1,cnt[v]++,q.push(v);
17                if(cnt[v]>n)return 1;
18            }
19    }
20    return 0;
21 }
```

d的初始化
重要吗

对于判负环不重要

所有点都入队
都有资格更新

SPFA判负环 - 超级源

所有点都入队
都有资格更新



增加超级源S
从S连向所有点
边权为0

现场挑战

944

请同学概括题目大意
已知什么求什么

写出图论要素

限时5分钟

944图论要素识别

有向图

负权图

多源单汇

单源多汇

可能不连通

判负权回路

汇点不连通点
都忽略

判负权回路和
和汇点连通

$n \leq 2000$, $m \leq 100000$, 稀疏图

算法选择

SPFA

单点入队

Bellman-Ford

怎么做

Bellman-Ford错误代码

```
8 int BellmanFord(){
9     fill(dst,dst+n+9,INF);
10    dst[1]=0;
11    for(int k=1;k<=n-1;k++)
12        for(int i=1;i<=n;i++)
13            for(int j=0;j<to[i].size();j++)
14                dst[to[i][j]]=min(dst[to[i][j]],dst[i]+w[i][j]);
15    for(int i=1;i<=n;i++)
16        for(int j=0;j<to[i].size();j++)
17            if(dst[to[i][j]]>dst[i]+w[i][j])return -1;
18    return 0;
19 }
```

错在哪?

```
31 if(BellmanFord()==-1){cout<<-1<<endl;return 0;}

32 int ans=0;
33 for(int i=1;i<=n;i++)
34     if(dst[i]<INF)ans+=dst[i]*x[i];
35 cout<<ans<<endl;
```

不连通的点
也都在更新

更新前要判断
源点与该点连通性

易错点总结

判断负环



不连通

SPFA

不连通点不入队
无法更新找负环

Bellman-Ford

不连通点
也能更新找负环
但d标记不是INF

差分约束系统

$$\begin{aligned}x_2 - x_1 &\leq 3 \\x_3 - x_2 &\leq -2 \\x_1 - x_3 &\leq -2\end{aligned}$$

一定无解
为什么?

差分	两项做差
----	------

约束系统	不等式组
------	------

$$x_b - x_a \leq c$$

差分不等式



$$x_b \leq x_a + c$$

对
应



有向边连接
两个节点

x_u 对应源点到 u 的最短路

差分约束系统 - 建约束图

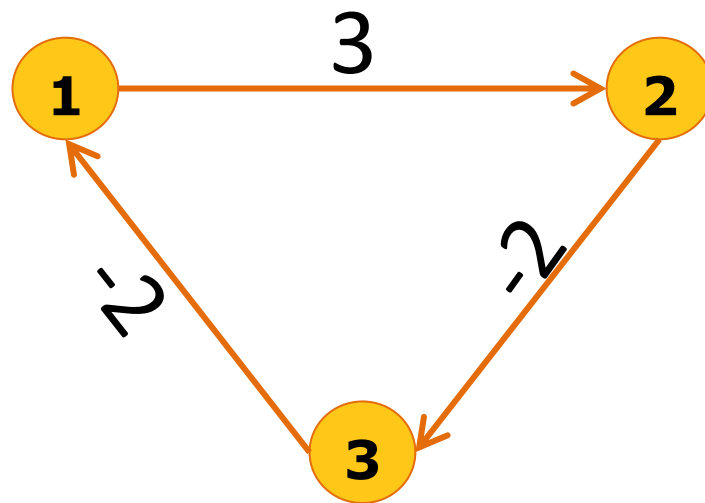
$$\begin{aligned}x_2 - x_1 &\leq 3 \\x_3 - x_2 &\leq -2 \\x_1 - x_3 &\leq -2\end{aligned}$$



$$\begin{aligned}x_2 &\leq x_1 + 3 \\x_3 &\leq x_2 - 2 \\x_1 &\leq x_3 - 2\end{aligned}$$

差分约束
系统无解

约束图有
负权回路



差分约束系统 - 建约束图

差分约束
系统无解

当且
仅当

约束图有
负权回路

请同学写出证明过程

限时5分钟

证明思路

1

约束图有
负权回路



差分约束
系统无解

2

差分约束
系统无解



约束图有
负权回路

这两个方向哪个更容易证?

无中生有!难!

证明思路

1

约束图有
负权回路



差分约束
系统无解

存在负权回路 u_1, u_2, \dots, u_k

$$w(u_1, u_2) + w(u_2, u_3) + \dots + w(u_k, u_1) < 0$$



存在以下k个差分约束

$$\begin{aligned} & x_{u_2} - x_{u_1} + x_{u_3} - x_{u_2} + \dots + x_{u_1} - x_{u_k} \\ & \leq w(u_1, u_2) + w(u_2, u_3) + \dots + w(u_k, u_1) < 0 \end{aligned}$$

证明思路

2

差分约束
系统无解



约束图有
负权回路

逆否
命题

约束图无
负权回路



差分约束
系统有解

哪个命题更容易证?

如何证明?

现场挑战

959

请写出题目大意
已知什么求什么

写出算法步骤和复杂度
或者写出难点

限时3分钟

快快编程作业

944

647

959

整理本课定理证明发课程群

拓展题

943,960