最短通路问题

离散数学—图论初步

南京大学计算机科学与技术系

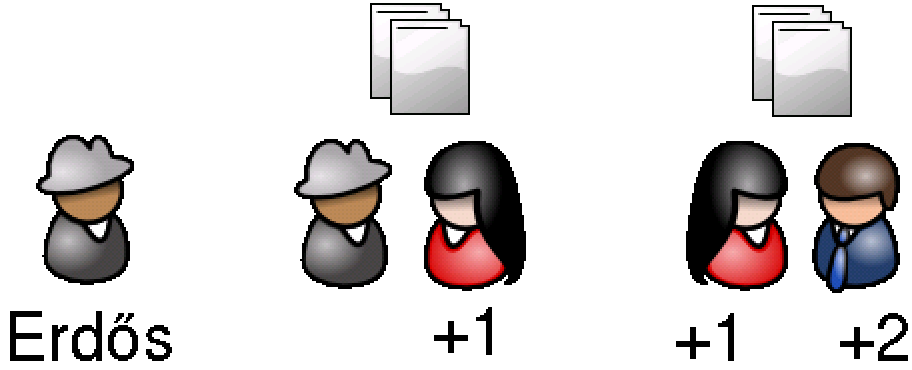
内容提要

- 引 言
- **Dijkstra算法**
- 旅行商问题（**TSP**）

埃德斯数（Erdős number）

Paul Erdös (1913-1996), Hungary, U.S.A., Israel

Erdős number



带权图与最短通路问题

- 带权图：三元组 (V, E, W)，(V, E)是图，W是从E到非负实数集的一个函数。$W(e)$表示边e的权。
- 一条通路上所有边的权的和称为该通路的长度。
- 两点之间长度最小的通路称为两点之间的最短通路，不一定是唯一的。
- 单源点最短路问题

  给定带权图 $G(V, E, W)$，并指定一个源点，确定该源点到图中其它任一顶点的最短路（长度和路径）。

Dijkstra最短路径的算法思想(1959)

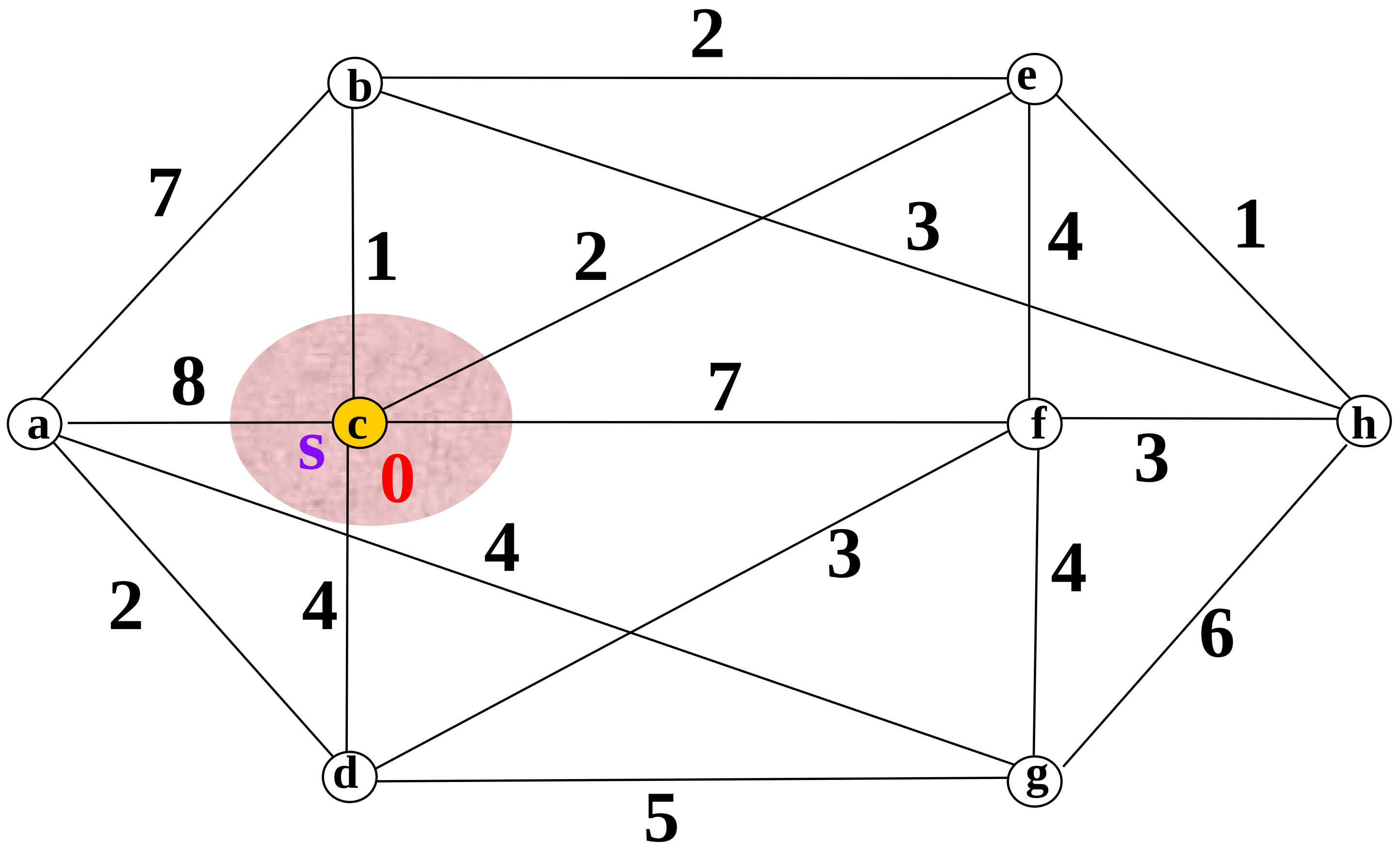
- 源点s到顶点v的最短路径若为s…uv，则s…u是s到u的最短路径。
- (n-1)条最短路径按照其长度的非减次序求得，设它们的相应端点分别为u_1，…u_{n-1}，最短路径长度记为$d(s, u_i)$， $i=1,\ldots n-1$.
- 假设前i条最短路径已知，第(i+1)条最短路径长度：

$$d(s, u_{i+1})=\min\{d(s, u_j) +W(u_j, u_{i+1}) \mid j=1,\ldots i\}$$

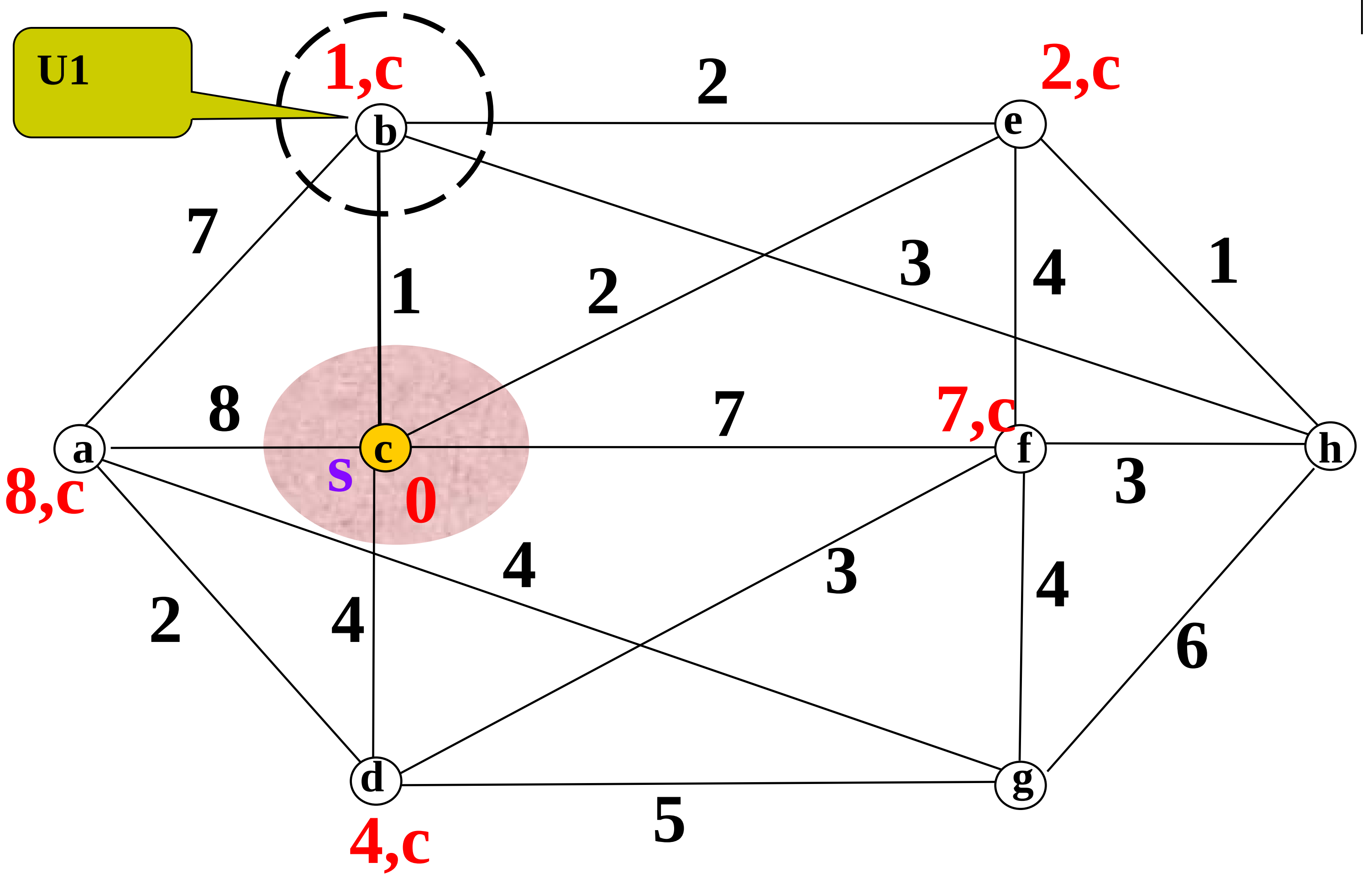
求最短路径的Dijkstra算法

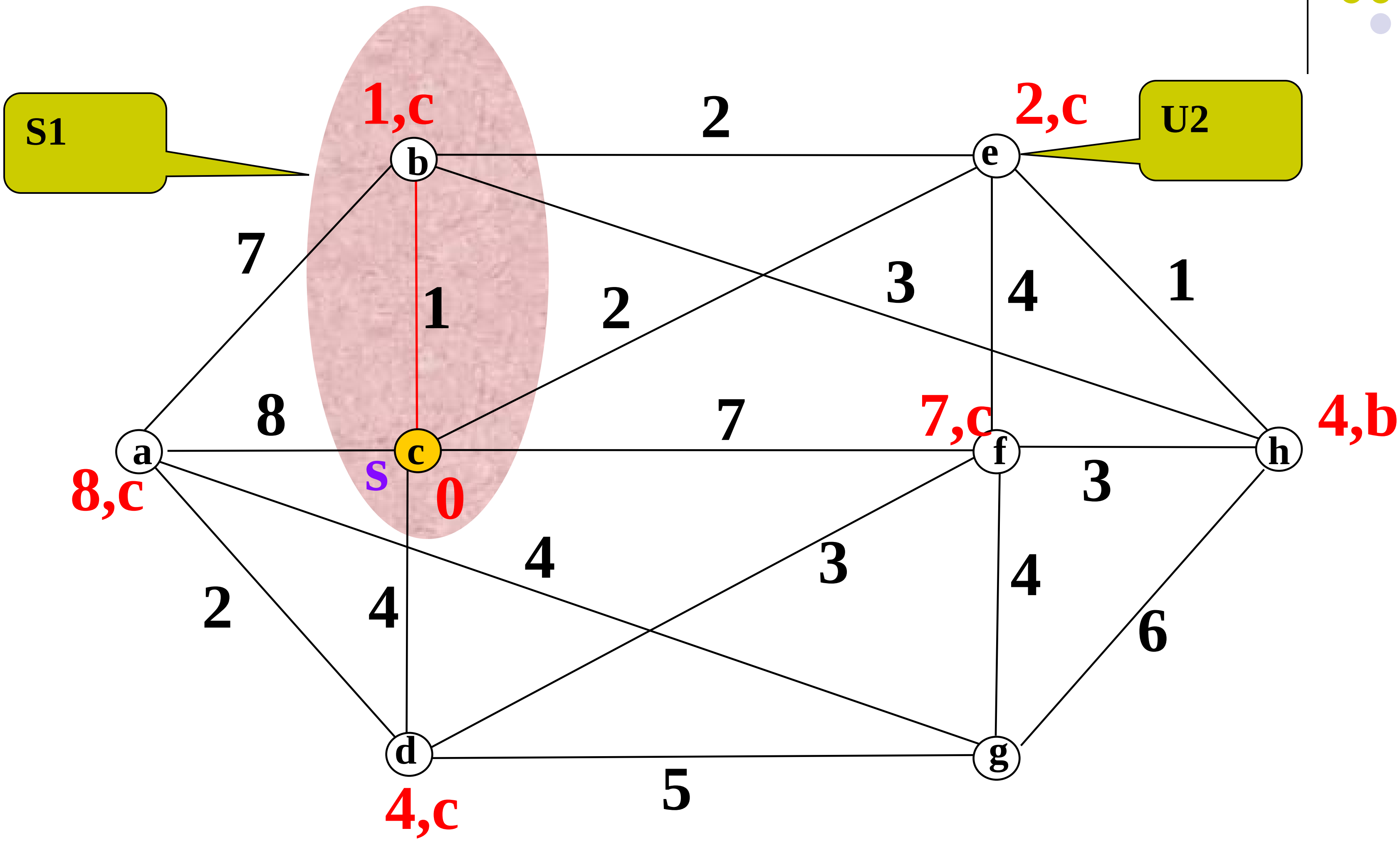
- 输入：连通带权图G，$|V_G|=n$, 指定顶点$s \in V_G$
- 输出：每个顶点v的标注$(L(v), u)$, 其中：
  - $L(v)$即从s到v的最短路径长度（目前可得的）
  - u是该路径上v前一个顶点。

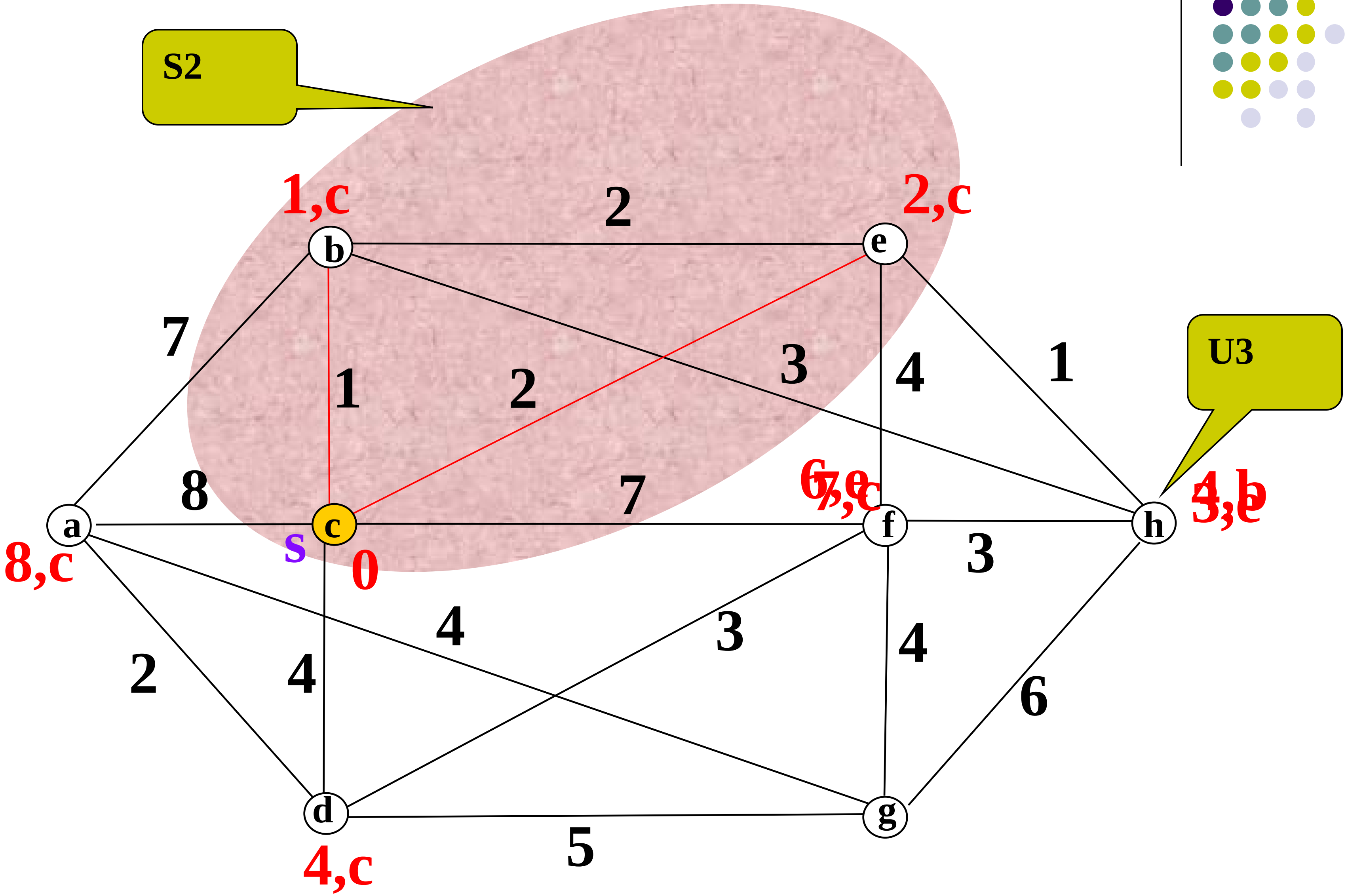
求最短路的一个例子

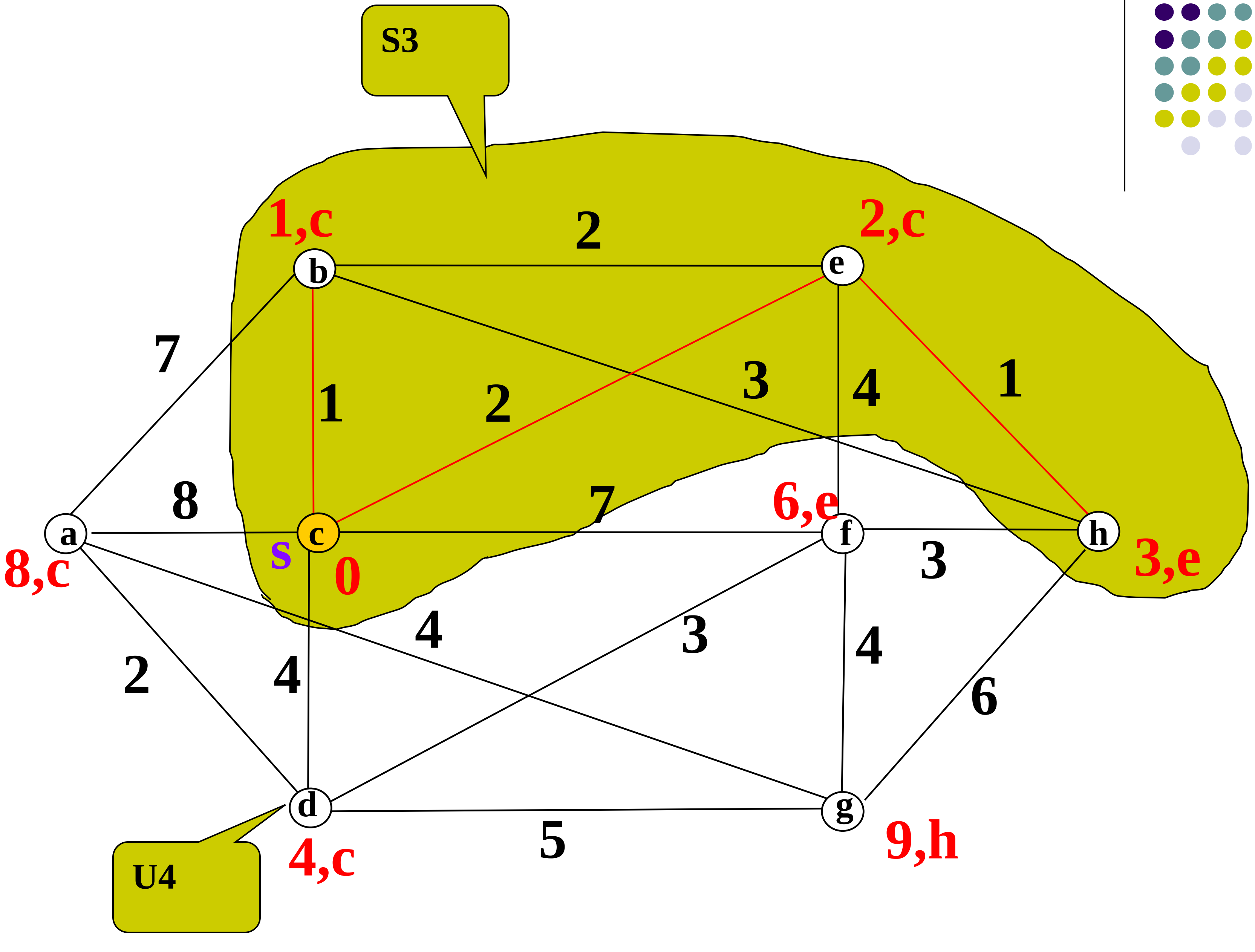


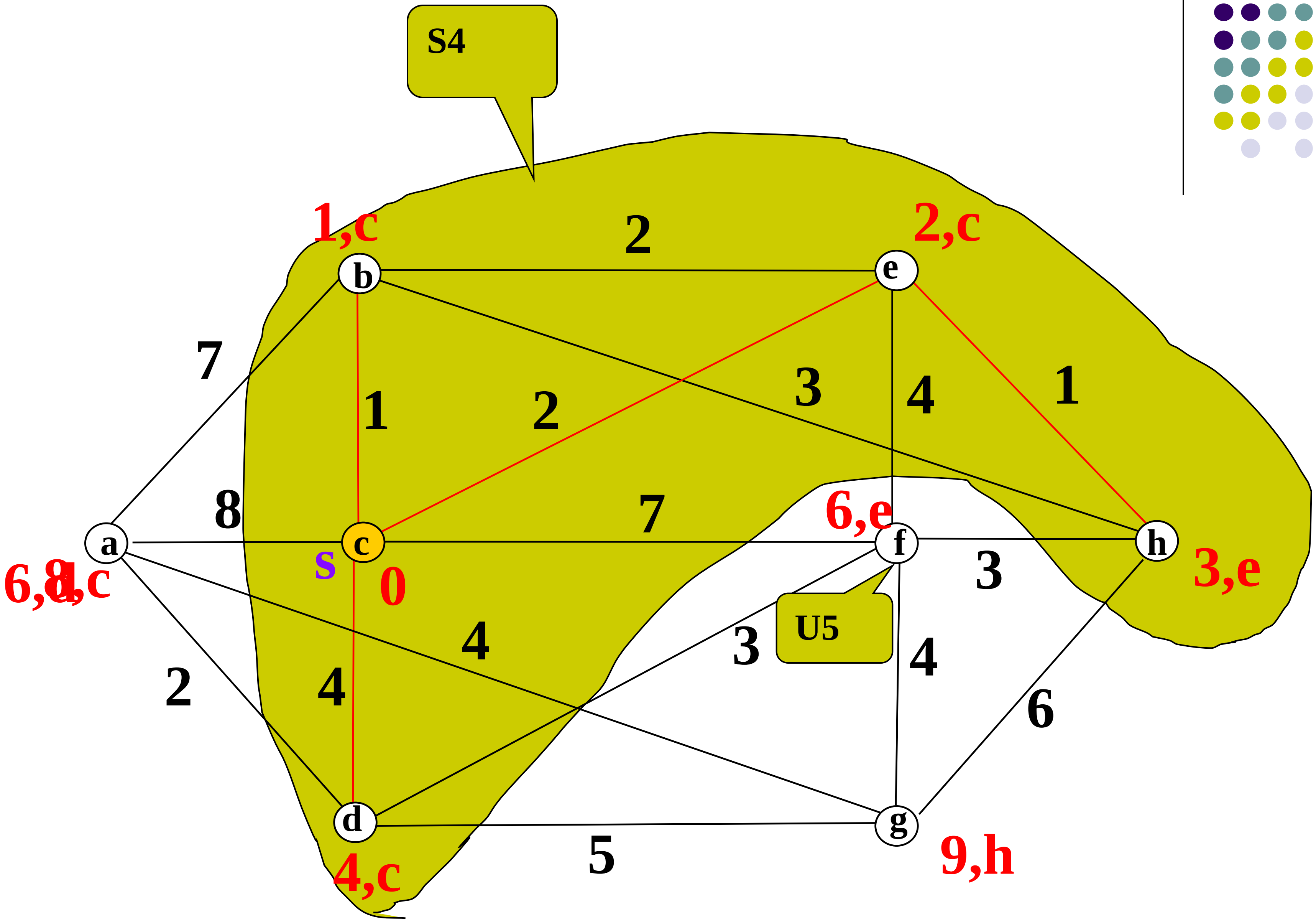
求最短路的一个例子



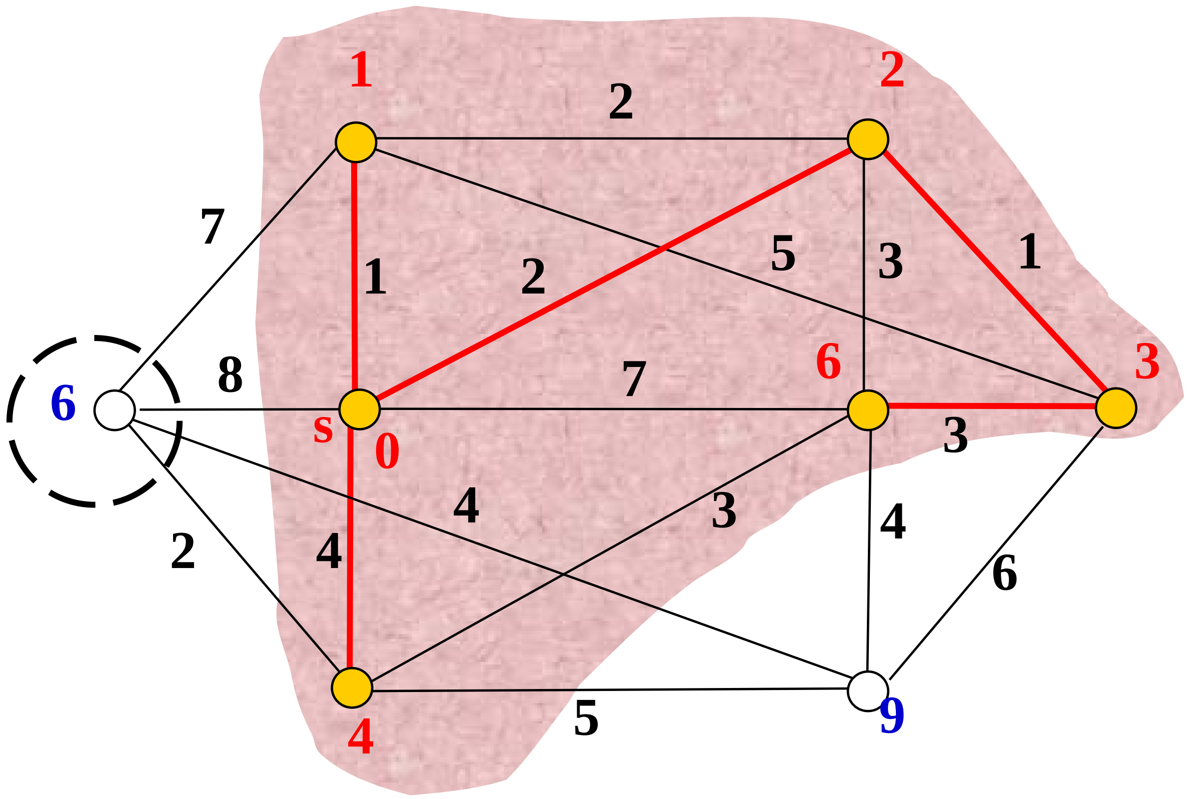

S1
U2
1,c
2,c
b
e
a
c
f
h
d
g
s
0
8,c
7,c
4,b
4,c
2
7
1
2
3
4
1
8
7
3
4
3
2
4
4
6
5


S2
U3
1,c
2,c
b
e
2
7
1
2
3
4
1
8
7
6,e
7,c
4,b
3,e
a
c
f
h
S
0
8,c
3
4
3
4
2
4
6
d
g
5
4,c

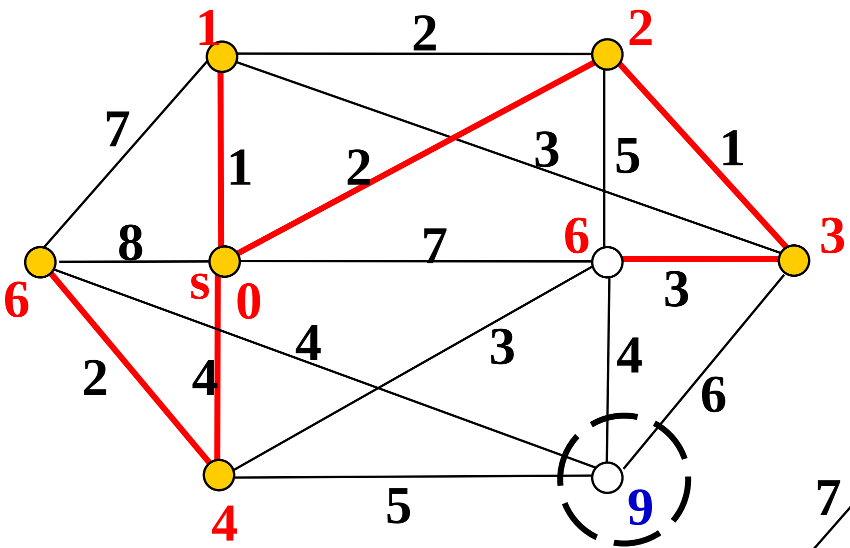

S3
1,c
2
2,c
b
e
7
1
2
3
4
1
8
7
6,e
a
c
f
h
8,c
s
0
3
3,e
4
3
4
2
4
6
d
g
5
4,c
9,h
U4

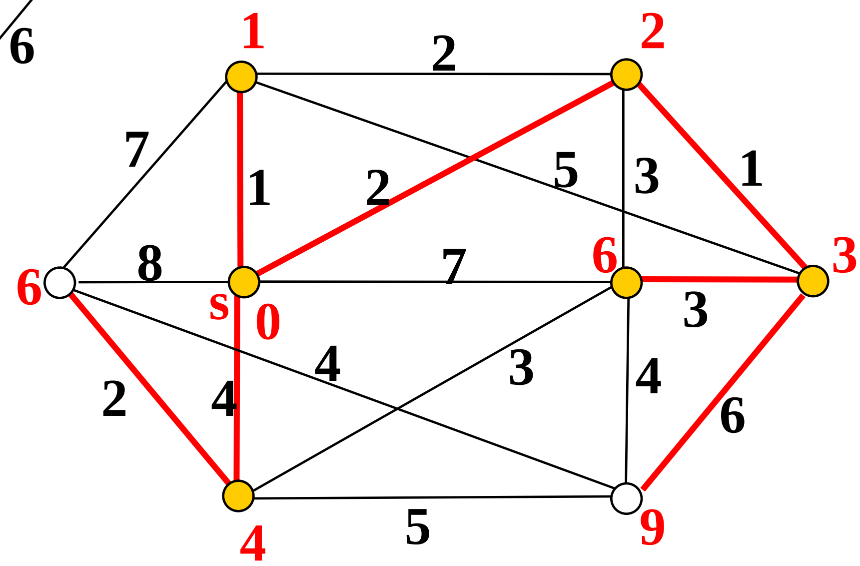

S4
1,c
2,c
2
b
e
7
1
2
3
4
1
8
7
6,e
a
c
f
h
s
0
3
3,e
U5
4
3
4
2
4
6
d
g
5
4,c
9,h

求最短路的一个例子(续)



求最短路的一个例子(续)


1
2
2
7
1
2
3
5
1
8
7
6
3
6
s
0
3
4
3
4
2
4
6
5
4
9


1
2
2
7
1
2
5
3
1
8
7
6
3
6
s
0
3
4
3
4
2
4
6
5
4
9

Dijkstra算法的描述

1．初始化：i=0, $S_0=\{s\}$, $L(s)=0$, 对其它一切$v \in V_G$, 将$L(v)$ 置为∞。若$n=1$，结束。

2．$\forall v \in S_i{}'=V_G-S_i$, 比较$L(v)$和$L(u_i)+W(u_i, v)$的值 ($u_i \in S_i$)

如果$L(u_i)+W(u_i, v)<L(v)$, 则将v的标注更新为$(L(u_i)+W(u_i, v), u_i)$，

即：$L(v)=\min\{ L(v), \min_{u \in S_i}\{L(u)+W(u,v)\} \}$

3. 对所有$S_i{}'$中的顶点，找出具有最小$L(v)$的顶点v, 作为u_{i+1}

4. $S_{i+1} = S_i \cup \{u_{i+1}\}$

5. i = i+1; 若i=n-1, 终止。否则：转到第2步。

Dijkstra算法的分析

- 可终止性
  - 计数控制
- 正确性

  需证明当算法终止时

  - $L(v)=d(s, v)$对一切v成立。
  - 由标记中的诸u_i确定的路径是一条最短路径

    ($这里d(s,v)$是s到v的最短路径长度，即距离。)
- 复杂性
  - $O(n^2)$

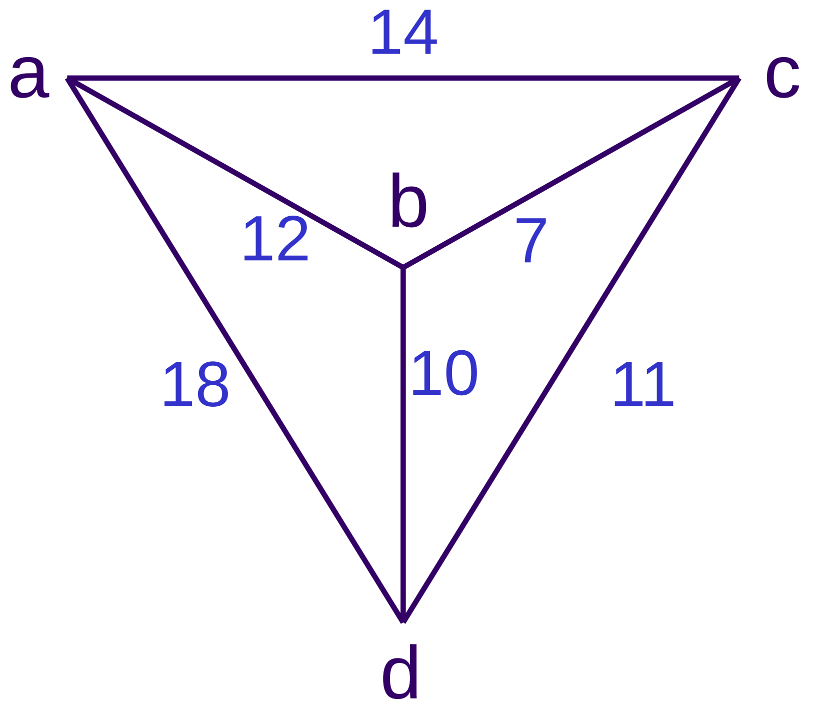
旅行商问题
(Travelling Salesman Problem, TSP)

- n个城市间均有道路，但距离不等，旅行商从某地出发，走过其它n-1城市各一次，最后回到原地，如何选择最短路线？
- 数学模型：
  - 无向带权图G：顶点对应于城市，边对应于城市之间的道路，道路长度用相应边的权表示。
  - 问题的解：权最小的哈密尔顿回路。
  - G是带权完全图，总共有(n-1)!/2条哈密尔顿回路。因此，问题是如何从这(n-1)!/2条中找出最短的一条。

(含25个顶点的完全图中不同的哈密尔顿回路有约3.1×10^{23}条，若机械地检查，每秒处理10^9条，需1千万年。)

旅行商问题

- 一个货郎（销售员）生活在城市a，假定访问的城市是d, b, c，然后回到a，求完成这次访问的最短路径的距离.



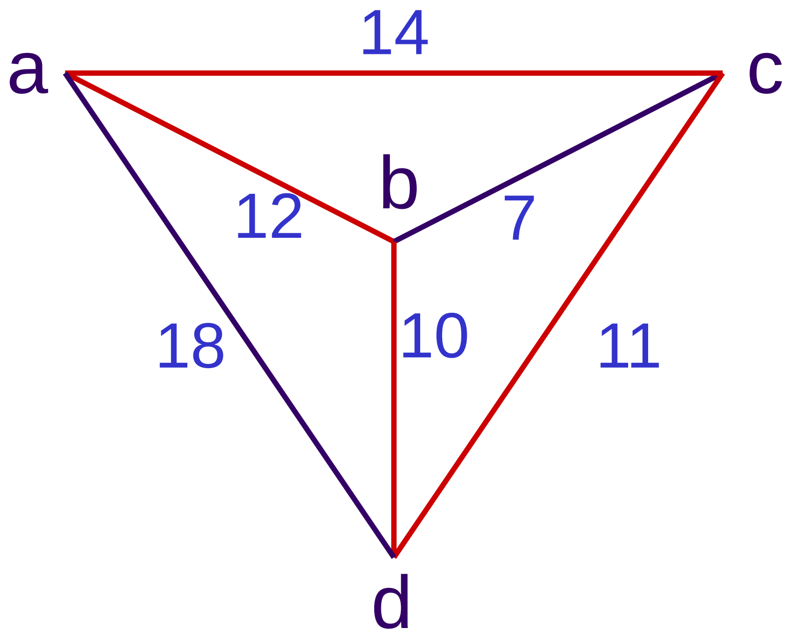
旅行商问题

- 解：列出哈密尔顿回路, 并求其距离：

（1）（abcda）=（12+7+11+18）= 48

（2）（acbda）=（14+7+10+18）= 49

（3）（abdca）=（12+10+11+14）= 47



旅行商问题

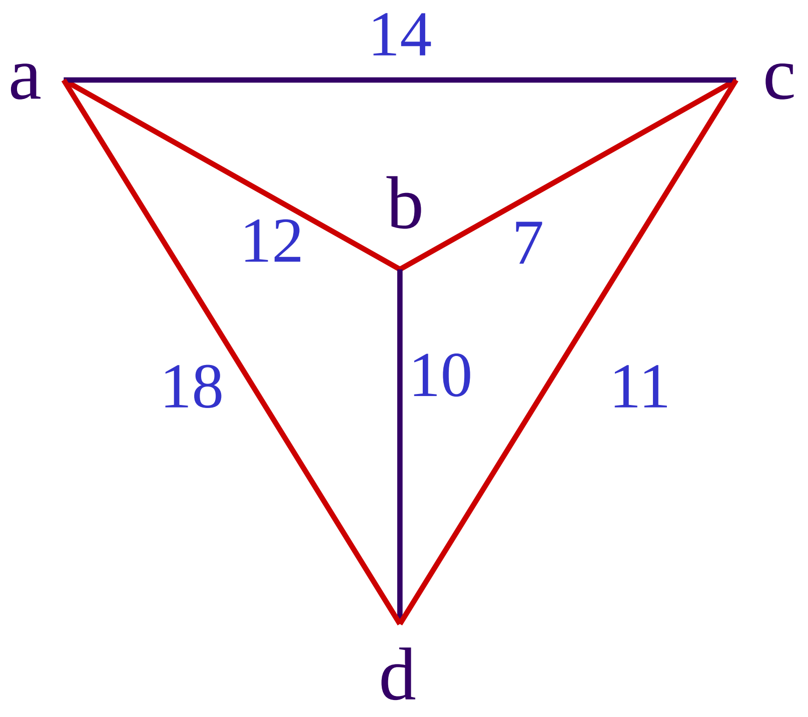
- 哈密尔顿回路（路径）的最短路径问题！
- 下面介绍一种最邻近算法：

（1）选择任一顶点作为始点，找出离始点距离最小的顶点，形成一条边的初始路径；

（2）设u是最新加到这条路径上的顶点，从不在这条路径上的所有顶点中选择一个与u距离最小的顶点, 把连接u与此结点的边加入路径中；重复执行直到G中的各顶点均含在这条路径中。

旅行商问题

（3）把始点到最后加入的顶点的边放入路径中得到一条哈密尔顿回路，并为近似最短的哈密尔顿回路.



旅行商问题(TSP)的研究进展

- （在最坏情况下）时间复杂性为多项式的算法?
- （在最坏情况下）时间复杂性为多项式的近似算法
  - 保证: $W \le W' \le cW$ (c=3/2), 误差为50%
- 实际应用中，已有好的算法能够在几分钟内处理1000个节点的规模，误差在2%。

作业

- 教材（第六版）[9.6]
  - p.507: 6, 21, 22, 25
- 教材（第七版）[10.6]
  - p.605: 6, 21, 22, 25