算法第9次作业

PB18000227 艾语晨

1

2-SAT from WikiPedia

将原图的每一个x拆分成两个点，分别代表他为假和真（比如将他们标号为2x和2x+1）。将逻辑关系的约束条件表示为点之间代表推导的有向边。比如$x \vee y$，可以表示为$2x \rightarrow 2y+1$和$2y \rightarrow 2x+1$

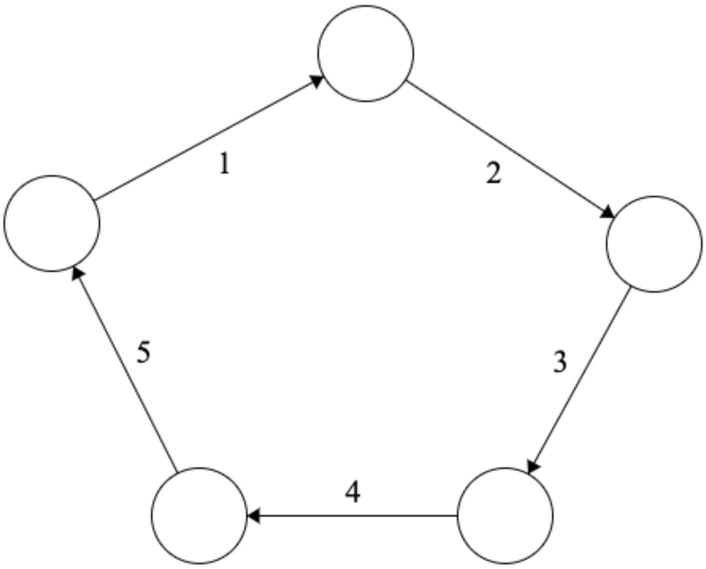
将原图利用Tarjan算法缩点，如果对于一个点xx的两个状态真或假都在一个强联通分量（scc）中，那么肯定不成立。然后我们可以用*拓扑序*来染色求解，对于一个并未染色的点我们将它染成1，它的一个对立状态染成2。（此处应该是 **反向拓扑** ）然后不断重复。最后所有为1的就可以当做一组解了。

时间复杂度为O(V+E)

2

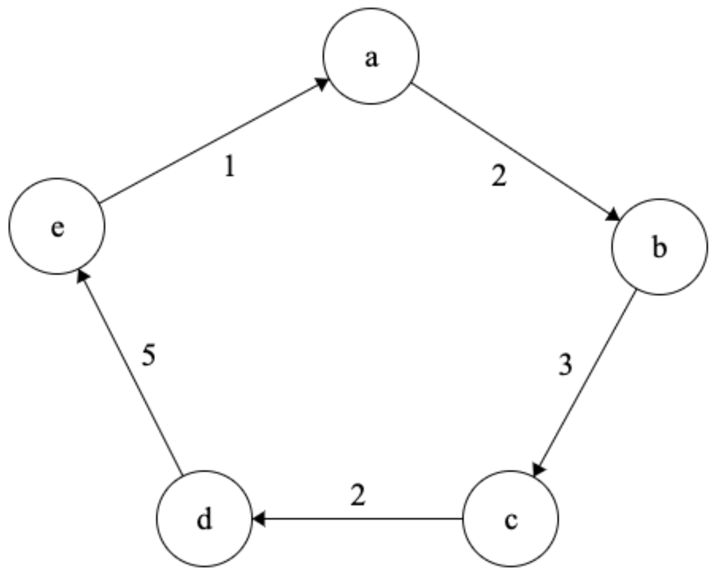
绘图网页只能连箭头，请忽略边的方向

- a



两集合均为除边 1 以外的边

- b



$S_G = \{ab, bc, de\}$

$T_G = \{ab, bc, cd, de\}$

- c

由Kruskal算法的原理可知，当生成最大生成树时，必定选取了每一个顶点具有最大边权的边，故$S_G \subseteq T_G$

- d

由上一问结论可知，如果$\omega(S_G) < \omega(T_G)$，那么一定是因为$S_G \subset T_G$。这时，一定会有一些顶点对共用一条边作为最大邻边。所以$|S_G| \geq |T_G|/2$，当且仅当所有顶点被两两成对时取等号。又由于T_G中剩下的边，其权重不大于此边相邻的顶点在S_G中选择的边，故$\omega(S_G) \geq \omega(T_G)/2$

- e

类似于Kruskal算法。

按权重对G的边排序，设A时最大生成树的边集，初始化为空集。加入最大权值的边。然后，若是排序中下一个边和已有边不成环，加入A，重复，直到对所有边进行过判断。若A中边的个数不是n-1，G不连通。否则A即为所求最大生成树。总时间为O(V+E)