题目 1

服务器上有 n 个请求等待处理。已知请求 i 的处理时长为 t_i。当服务器处理某请求时，其它尚未处理的请求需要等待，每个请求的等待时间定义为在它之前的请求的处理时长之和。请设计算法计算最优的处理顺序，使得所有请求的等待时间之和最小。

tree edge

不必讲了吧？ 3.(2) DFS on undirected graph: tree edge + back edge 证明任两点都可达 (从上到下, 从下到上)

Def: (Open) Ear (2-connected graph ear decomposition)

(closed) Ear (2-edge-connected graph closed-ear decomposition)

back-edge

题目 2

给定带权无向连通图 $G = (V, E, w)$ (w 为权重函数) 与 G 中任一条边 $e = (u, v) \in E$。

证明: e 不属于 G 的任何最小生成树当且仅当 u、v 之间存在由权重均小于 $w(e)$ 的边构成的路径。

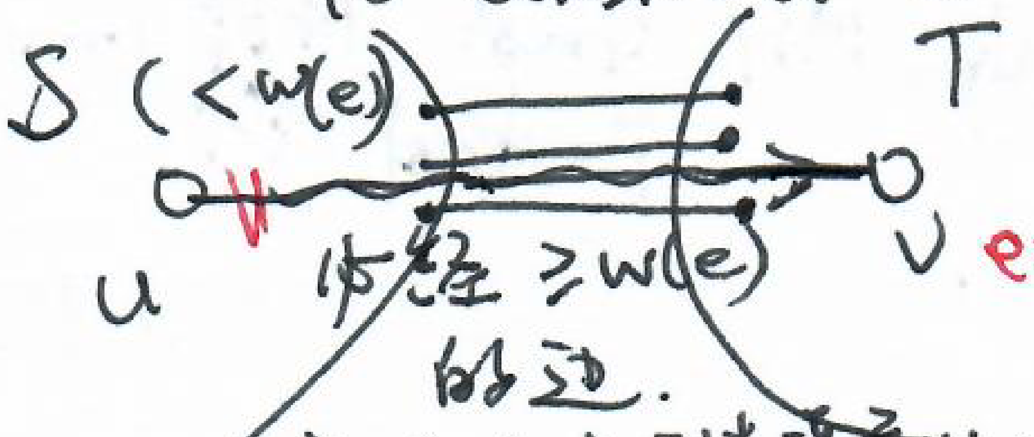
原题: Give an algorithm to decide whether e is contained in an MST of G.

(如何解) $O(n+m)$

$\Leftarrow$ By cycle property.

(1) deleting edges $> w(e)$; (2) testing connectivity.

$\Rightarrow$ By contradiction. To construct a cut.



By cut property.

$e=(u,v) \in E$.

$\Rightarrow$ $<w(e)$ 无法连通 u, v.

题目 3

给定无向连通图 $G = (V, E)$。

定义 (SCC 定向) 如果能对 G 的每条边确定一个方向，使得定向后的有向图是强连通的，则称 G 存在 SCC 定向。

$\boxed{\text{Robbin's Thm (Herbert Robbin) (1939)}}$

An undirected graph G has strong orientations iff G is 2-edge-connected ~~edges~~ graphs.

(1) 请证明 G 存在 SCC 定向当且仅当 G 中没有桥。

(1) Pf. "$\Rightarrow$": By contradiction.

(2) 假设 G 中没有桥，请设计一个线性时间算法给出 G 的一种 SCC 定向。"$\Leftarrow$"

uv lies on a cycle

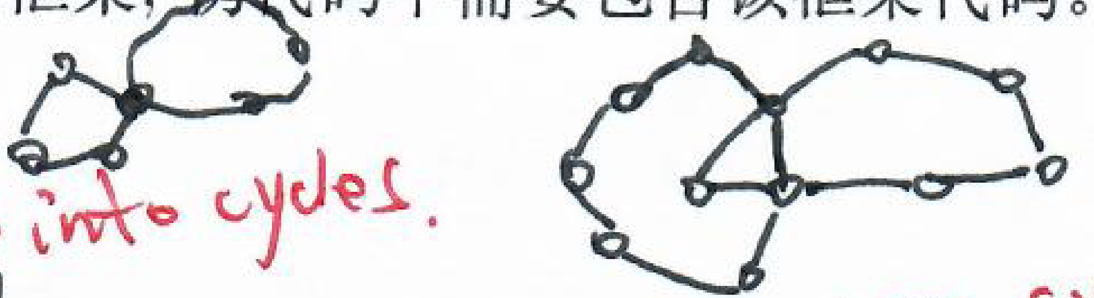
Hierholzer's

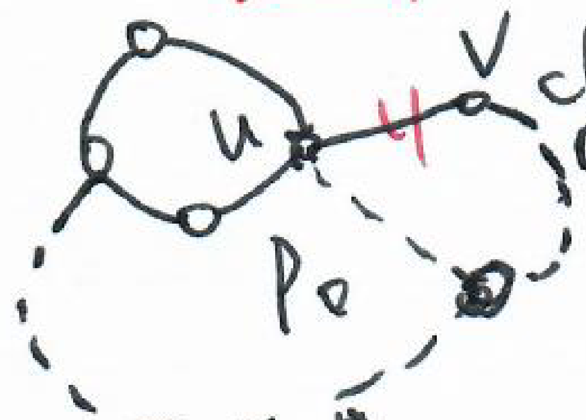
要求: 如果你使用了图遍历框架，伪代码中需要包含该框架代码。

与欧拉图关系 (1912, Veblen's thm)

Eulerian graph iff decomposes into cycles.

(不是)





closed ear

(open) ear

Def (open) Ear:

closed Ear:

一条 uv, xy lie on a common cycle.

An ear of a graph G is a path that is maximal w.r.t. internal vertices having degree 2 in G and is (closed-ear decomposition)

题目 4 正负

SSBP ; APBP (complexity) (CLRS 中的 path 定义)

考虑带权无向连通图 G。

简单路径 (simple path) 还是允许有 cycle?

(思考题)

定义 (瓶颈长度) G 中路径的瓶颈长度指的是该路径上最小的边权重。

(简单路径? simple path? negative cycles?)

定义 (瓶颈距离) G 中两点的瓶颈距离指的是该两点间所有路径中的最大瓶颈长度。

表示初始化 ① 有边: $w(e)$ ② 无边: $-\infty$

单源

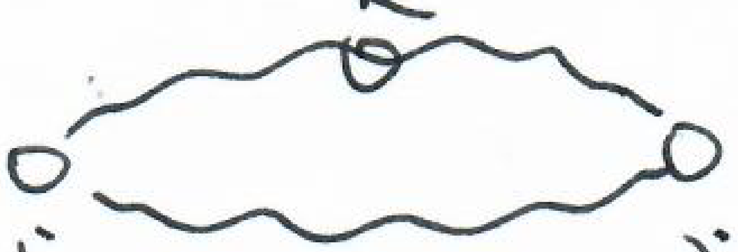
请设计算法求 G 中所有点对之间的瓶颈距离 (不必给出具体路径)。

(Your goal is to, for any s and t, find a path from s to t whose minimum edge weight is maximized.)

要求: 时间复杂度为 $O(n^3)$ (n 为 G 的顶点数)。请写出伪代码。

为什么 APBP 中没有该问题? (看最小边) ① 小于最小的 ② 大于最小的

```
For k from 1 to n
    For i from 1 to n
        For j from 1 to n
            B[i,j] ← max(B[i,j], min(B[i,k], B[k,j]))
```



$\boxed{\text{Pf?}}$

问题在于: 该 DP 依赖于 (没有负权 cycle!!! 在 APSP 中)

(Thm 4.2.21 of West)

The connectivity of G $\kappa(G) \triangleq$ the maximum k s.t. $\boxed{\lambda(x,y)} \geq k$ for

结论: $\kappa(G) \leq \lambda(G) \leq \delta(G)$ (✗) (Thm 5.1.1).

题目 5

(1) 请写出 Menger 定理 (无向图"点连通"或"边连通"版本)。 (Whitney 1932)

(2) 请使用 Menger 定理证明: 如果 G 是 3-正则图, 则 $\kappa(G) = \lambda(G)$。Thm (5.1.2).

(提示: 可使用教材中基于 Menger 定理证明的定理或推论, 但需明确说明。)

(2) 使用 Thm 5.1: (1) $\kappa(G) \leq \lambda(G)$

(2) 只需证 $\underline{\kappa(G) \geq \lambda(G)}$

$\forall x, y$: (1) $\exists \geq \lambda(G)$ pairwise edge-disjoint x-y paths. (2) G is 3-regular. these paths are pairwise internally-disjoint x-y paths. $\Rightarrow \kappa(G) \geq \lambda(G)$.

	conn.	edge-conn.
local	Thm 5.1.6	Thm 5.2.1
global	Thm 5.1.7	Thm 5.2.2

$(x,y) \notin E$. pf. $\lambda(x,y) = \lambda'(x,y)$. $\lambda(x,y) \geq \lambda$. $\kappa(x,y) = \kappa'(x,y)$

edge, un, di, local, global. (5.1.6 Thm).

(有向图 digraph 版本).

题目 6

中国象棋有"马走日"的规则。

证明或证伪: 对于任何 $n \in \mathbb{Z}^+$, 在 $4 \times n$ 的棋盘上, 不存在"马踏棋盘回路"(从某棋盘格开始, 遍历所有棋盘格一次且仅一次, 并回到起始棋盘格)。

(提示: 以 4×5 为例, 见图 1。)

txp PPT: 4×4 ✗ 5×5 ✗

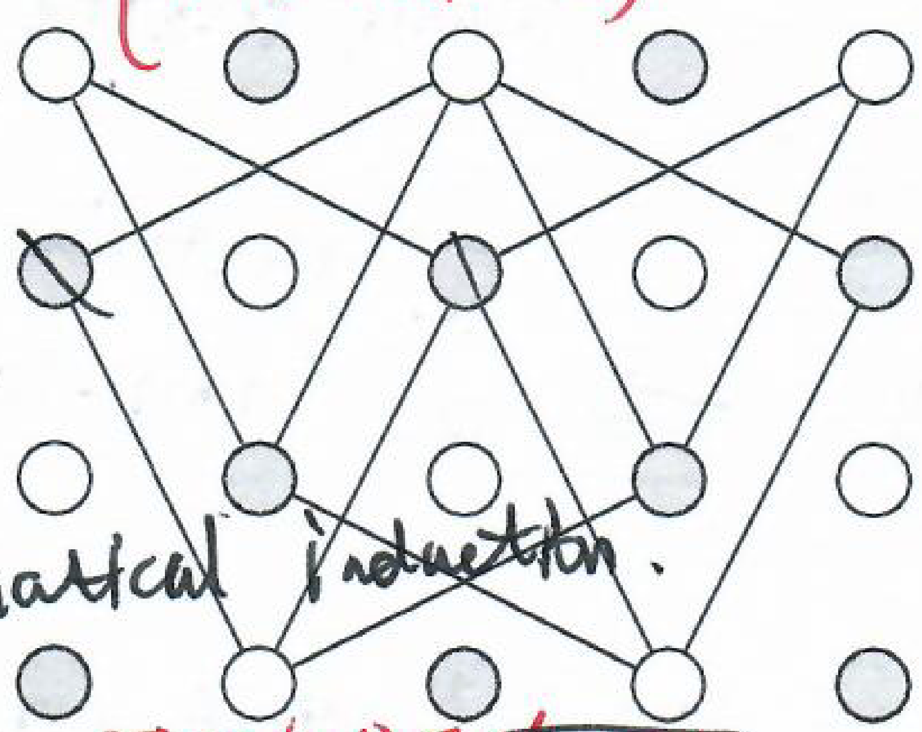
("方向自由"步到不合法)

原题: 证明"不存在性"。 (3-10: 11月20日)

Thm Name?

If G has a Hamiltonian cycle, then for each nonempty set $S \subset V(G)$, the graph $G - S$ has at most $|S|$ components.

pf.



~~只需找一个定理~~.

① 第一反应: By mathematical induction.

② 使用定理. (必要条件)

需要找是平板状子.

图 1: 灰白相间的棋盘。

(1) 删除中间两排 $\Rightarrow G$: $\kappa(G - S) = |S|$.

(2) 删除中间两排的灰色节点. $|S| = 5$

$\kappa(G-S) \geq 5 + 1$ (其余点).

$\boxed{\kappa(G-S) \leq |S|}$ $\forall \emptyset \neq S \subseteq V$.

$|S| \geq \kappa(G-S)$

tough (坚韧性)

题目 7 (树(Spanning Tree 游戏?) 作为思考题?)

考虑无向连通图 G 上的双人游戏。在该游戏中, 两位玩家交替选择顶点。游戏开始时, 先手玩家任选一个顶点。接下来, 每位玩家选择的顶点需要 (1) 与另一位玩家刚刚选择的顶点相邻 (2) 之前未被选过。(也就是说, 两位玩家选择的顶点构成一条简单路径。) 若无顶点可选, 则该玩家为输家, 另一玩家为赢家。

请解答并证明: 当 G 具有何种性质时, 后手玩家有必胜策略, 否则先手玩家有必胜策略。

原题: 证明"perfect Matching"时, ...

pf. (1) ⇒ ← 先手 ← 后手

(2) ⇐. 无 perfect matching, 要为先手构造策略

胜策略: 先手任取一个 maximal matching M. $v \notin M$.

A graph G is t-tough if $|S| \geq t\,k(G-S)$, $\forall S \subset V$.

Toughness of G is the max t s.t. G is t-tough.

Thm Petersen graph: 4/3 toughness.

题目 8 (附加题)

请设计一个线性时间算法判断有向图 G 是否含有有向奇圈。

v - x - y ... $\in M$. 先手 后手 先手

Conjecture: Chvátal's conjecture

argumenting path ⇒ 先手不会输 ⇒ 后手也不输 ⇒ 先手胜? t: some value of toughness suffices.

题目 1

服务器上有 n 个请求等待处理。已知请求 i 的处理时长为 t_i。当服务器处理某请求时, 其它尚未处理的请求需要等待, 每个请求的等待时间定义为在它之前的请求的处理时长之和。请设计算法计算最优的处理顺序, 使得所有请求的等待时间之和最小。

题目 2 undirected connected graph

给定带权无向连通图 $G = (V, E, w)$ (w 为权重函数) 与 G 中任一条边 $e = (u, v) \in E$。

证明: e 不属于 G 的任何最小生成树当且仅当 u、v 之间存在由权重均小于 $w(e)$ 的边构成的路径。

e does not belong to any MST of G

$\Leftrightarrow$ u, v can be joined by a path consisting entirely of edges that are cheaper than e.

Def: Ear (Open Ear): An ear of a graph G is a path that is maximal w.r.t. internal vertices having degree 2 in G and is contained in a cycle in G. An ear decomposition of G is a decomposition $P_0, \dots, P_k$ s.t. P_0 is a cycle and P_i for $i \geq 1$ is an ear of $P_0 \cup \dots \cup P_i$.

题目 3

给定无向连通图 $G = (V, E)$。

定义 (SCC 定向) 如果能对 G 的每条边确定一个方向, 使得定向后的有向图是强连通的, 则称 G 存在 SCC 定向。 (strong orientation)

Thm (Whitney 1932) A graph is 2-connected iff it has an ear decomposition.

Pf. $uv \in G - E(G_i)$, $xy \in E(G_i)$ lie on a common cycle.

(1) 请证明 G 存在 SCC 定向当且仅当 G 中没有桥。

SCC orientation. G has no bridge. G is bridgeless.

(2) 假设 G 中没有桥, 请设计一个线性时间算法给出 G 的一种 SCC 定向。

要求: 如果你使用了图遍历框架, 伪代码中需要包含该框架代码。

Def. Closed Ear: A closed ear in a graph G is a cycle C s.t. all vertices of C except one have degree 2 in G.

A closed-ear decomposition of G is a decomposition $P_0, \dots, P_k$ s.t. P_0 is a cycle and P_i ($i \geq 1$) is either an (open) ear or a closed ear in $P_0 \cup \dots \cup P_i$.

(Herbert Robbin) 1939. Thm: A graph is 2-edge-connected iff it has a closed-ear decomposition.

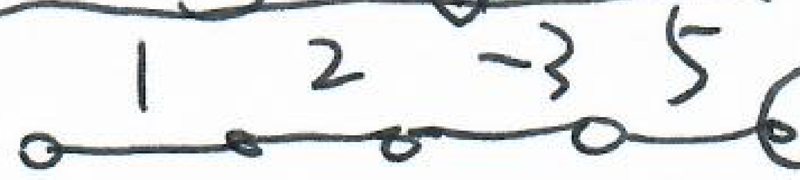
Pf. uv lies on a cycle C.

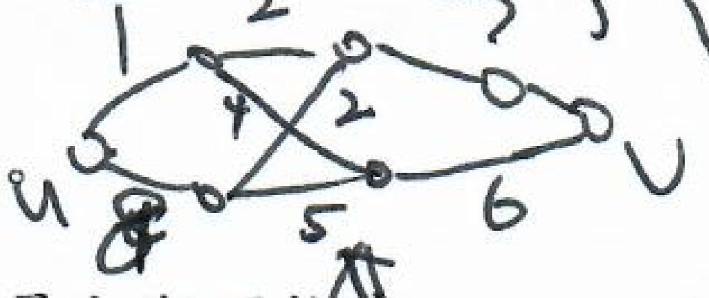
题目 4

考虑带权无向连通图 G。

bottleneck length

path





bottleneck length:

定义 (瓶颈长度) G 中路径的瓶颈长度指的是该路径上最小的边权重。

定义 (瓶颈距离) G 中两点的瓶颈距离指的是该两点间所有路径中的最大瓶颈长度。 bottleneck distance.

请设计算法求 G 中所有点对之间的瓶颈距离 (不必给出具体路径)。

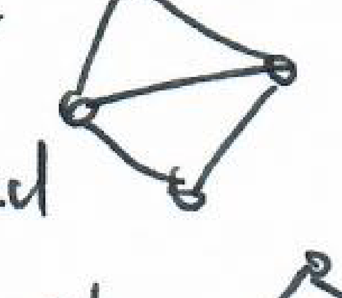
(Your goal is to, for any s and t, find a path from s to t whose minimum edge weight is maximized.)

要求: 时间复杂度为 $O(n^3)$ (n 为 G 的顶点数)。请写出伪代码。 (Veblen's Thm 1912)

Thm A graph is Eulerian iff. it can be decomposed into cycles.

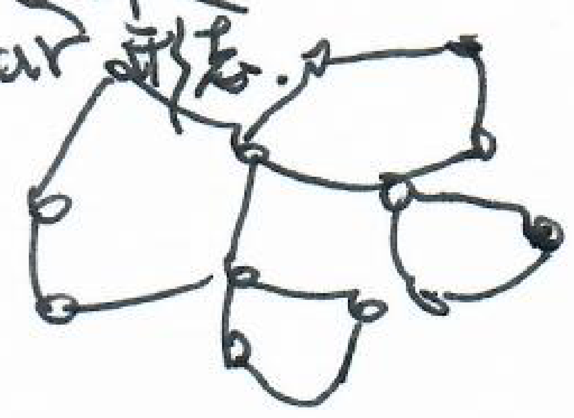
欧拉图 vs. 2-connected / 2-edge connected.

(1) Eulerian ⇒ 2-edge connected. ⇍ example



(2) Eulerian ⇏ 2-connected ⇍

注: 不一定非是 closed-ear 形态. ⇒ must be ⇒ 可能是



题目 1

服务器上有 n 个请求等待处理。已知请求 i 的处理时长为 t_i。当服务器处理某请求时，其它尚未处理的请求需要等待，每个请求的等待时间定义为在它之前的请求的处理时长之和。请设计算法计算最优的处理顺序，使得所有请求的等待时间之和最小。

题目 2

给定带权无向连通图 $G = (V, E, w)$ (w 为权重函数) 与 G 中任一条边 $e = (u, v) \in E$。

证明: e 不属于 G 的任何最小生成树当且仅当 u、v 之间存在由权重均小于 $w(e)$ 的边构成的路径。

题目 3

给定无向连通图 $G = (V, E)$。

定义 (SCC 定向) 如果能对 G 的每条边确定一个方向，使得定向后的有向图是强连通的，则称 G 存在 SCC 定向。

(1) 请证明 G 存在 SCC 定向当且仅当 G 中没有桥。

(2) 假设 G 中没有桥，请设计一个线性时间算法给出 G 的一种 SCC 定向。

要求: 如果你使用了图遍历框架，伪代码中需要包含该框架代码。

拓展: SSBP? $O(m)$? Binary search.

cycle 没有作用. 连通图. (STOC'07))

$\Rightarrow$ MST (maximum-spanning Tree)

~~taking the paths in~~

$O(n^2)$:

$O(E \lg V) \Rightarrow O(E + V \lg V)$ Priority Queue

题目 4 正/负 ~~Dijkstra~~ Floyd-warshall 算法不能直接应用到带负权的无向图上.

① 没有冲突考虑 cycle. cycle 时结果没有影响

② 初始值设置

$-\infty$ ×
$+\infty$ ×
∞ ×

降不下来

考虑带权无向连通图 G。

定义 (瓶颈长度) G 中路径的瓶颈长度指的是该路径上最小的边权重。

定义 (瓶颈距离) G 中两点的瓶颈距离指的是该两点间所有路径中的最大瓶颈长度。

请设计算法求 G 中所有点对之间的瓶颈距离 (不必给出具体路径)。

(Your goal is to, for any s and t, find a path from s to t whose minimum edge weight is maximized.)

要求: 时间复杂度为 $O(n^3)$ (n 为 G 的顶点数)。请写出伪代码。

初始化: $w(i,j) \begin{cases} \cancel{0}\ \boxed{-\infty} & i = j \\ e_{ij} & i \neq j \quad (i,j) \in E \\ \infty\ \boxed{-\infty} & i \neq j \quad (i,j) \notin E \end{cases}$

$k=1$ $i=2$ $j=2$

(2,2) $-\infty$ (2,1) 1 (1,2) 1

题目 5

(1) 请写出 Menger 定理 (无向图“点连通”或“边连通”版本)。

(2) 请使用 Menger 定理证明: 如果 G 是 3-正则图, 则 $\kappa(G) = \lambda(G)$。

(提示: 可使用教材中基于 Menger 定理证明的定理或推论, 但需明确说明。)

Undirected graph versions:

	conn.	edge-conn.
local	Thm 5.16 $\xRightarrow{pf}$	Thm 5.21
global	Thm 5.17 ($\Downarrow$ pf)	Thm 5.22 ($\Downarrow$)

digraph versions:

Thm 5.16: Let u and v be nonadjacent vertices in a graph G. ~~The~~ minimum # of vertices in a u-v separating set = maximum # of internally disjoint u-v paths in G.

Thm 5.17: A nontrivial graph G is k-connected ($k \geq 2$) iff for each pair u, v of distinct vertices of G, there are $\geq k$ interally-disjoint u-v paths in G.

Thm 5.21 For distinct vertices u, v in G, the minimum # of edges of G that separate u and v = the maximum # of pairwise edge-disjoint u-v paths in G.

Thm 5.22 A nontrivial graph G is k-edge-connected iff for each pair u, v of distinct ~~G contains~~ vertices of G, there are $\geq k$ pairwise edge-disjoint u-v paths in G.

题目 6

中国象棋有“马走日”的规则。

证明或证伪: 对于任何 $n \in \mathbb{Z}^+$, 在 $4 \times n$ 的棋盘上, 不存在“马踏棋盘回路”(从某棋盘格开始, 遍历所有棋盘格一次且仅一次, 并回到起始棋盘格)。

(提示: 以 4×5 为例, 见图 1。)

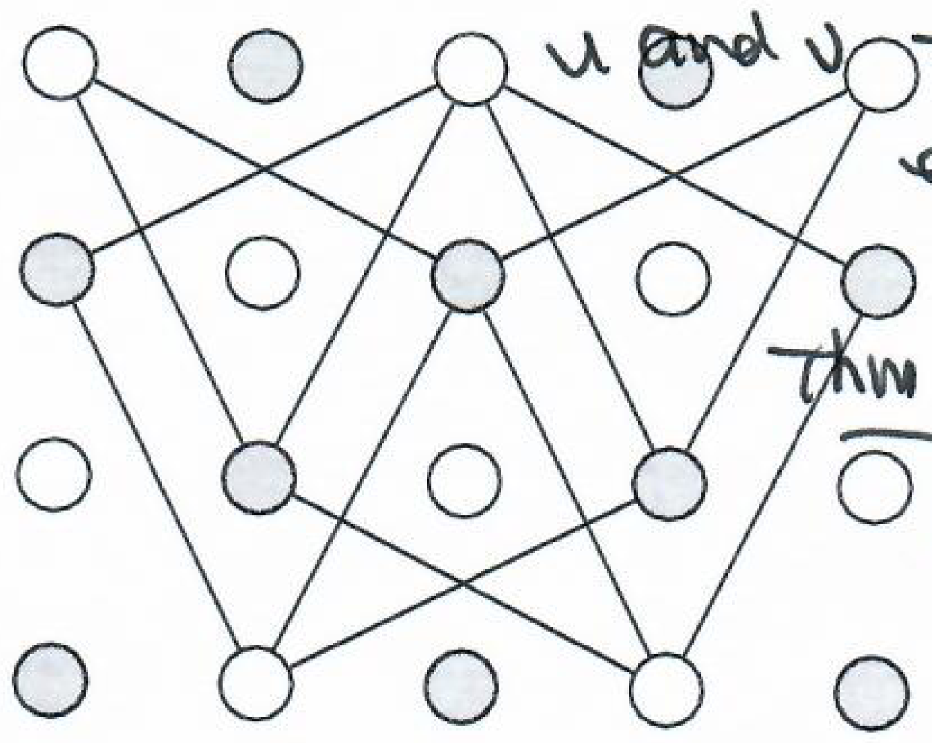


图 1: 灰白相间的棋盘。

已知 $\kappa(G) \leq \lambda(G)$

要证 $\kappa(G) \geq \lambda(G)$. Thm 5.22

根据 Thm 5.17, 构造 internally-disjoint paths $\Uparrow$ 3-regular.

Thm 5.21 edge-disjoint paths

题目 7

考虑无向连通图 G 上的双人游戏。在该游戏中, 两位玩家交替选择顶点。

游戏开始时, 先手玩家任选一个顶点。接下来, 每位玩家选择的顶点需要 (1) 与另一位玩家刚刚选择的顶点相邻 (2) 之前未被选过。(也就是说, 两位玩家选择的顶点构成一条简单路径。) 若无顶点可选, 则该玩家为输家, 另一玩家为赢家。

请解答并证明: 当 G 具有何种性质时, 后手玩家有必胜策略, 否则先手玩家有必胜策略。

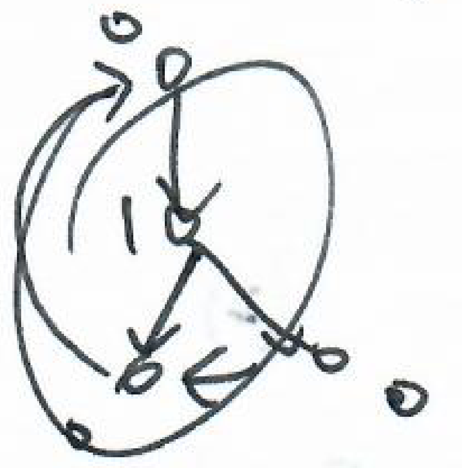
题目 8 (附加题)

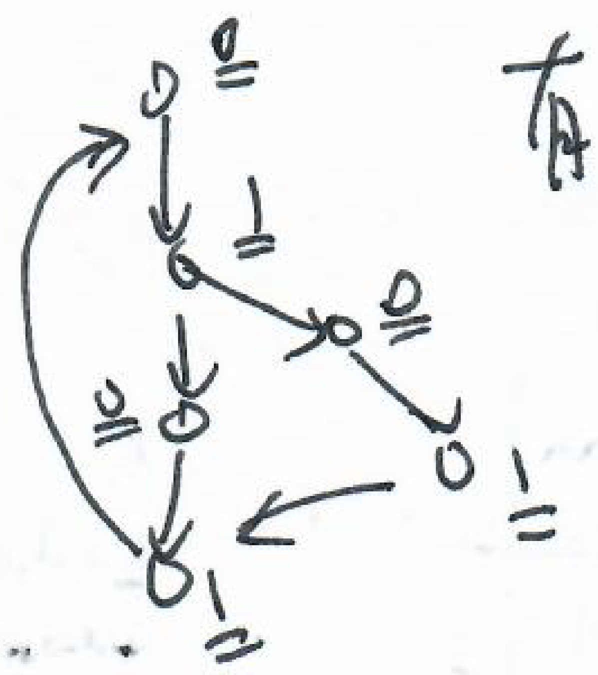
请设计一个线性时间算法判断有向图 G 是否含有有向奇圈。

判断有向图 G 是否含有有向奇圈.

一. 错误做法: DFS on G. + coloring.

$\boxed{\text{back edge}}$





检测出 ⇒ 必有

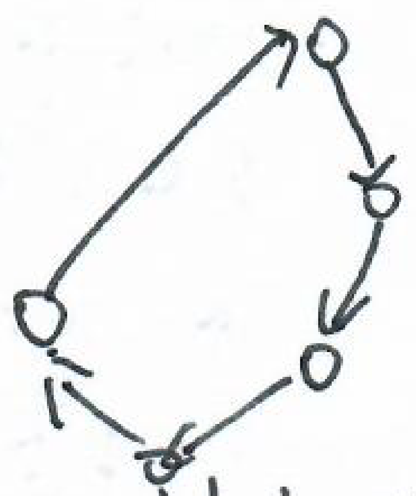
有 ⇒ 未必检测得出来.

一. 错误原因:

一. 正确定理.

$\boxed{\text{Thm}}$. A digraph G has an odd-length directed cycle iff one (or more) of its scc is non-bipartite. (when treated as an undirected graph.)

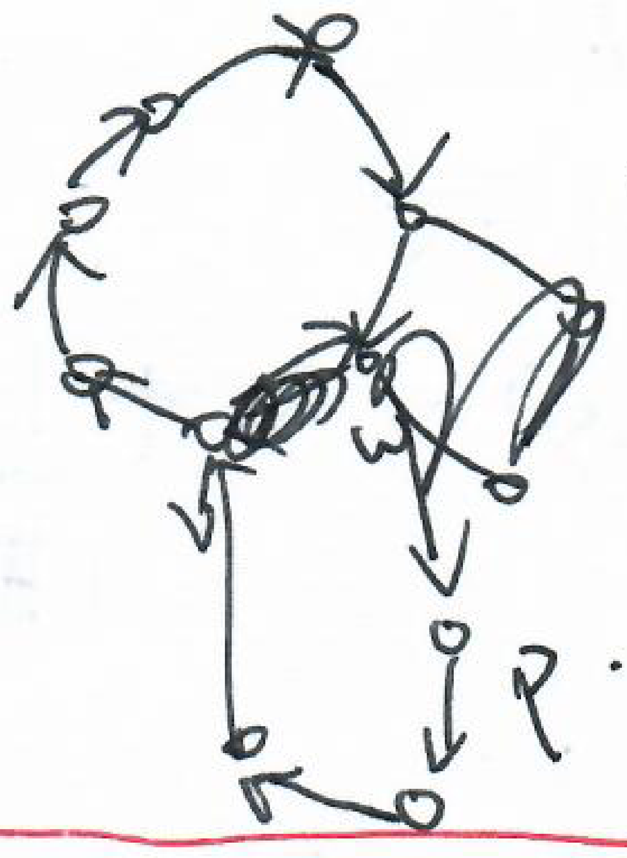
"⇒".



⇒ (1) contained in some scc
(2) non-bipartite.

odd-length directed cycle in G.

"⇐".



$\boxed{\text{an scc}}$

(1) C is a directed cycle ⇒ DONE.

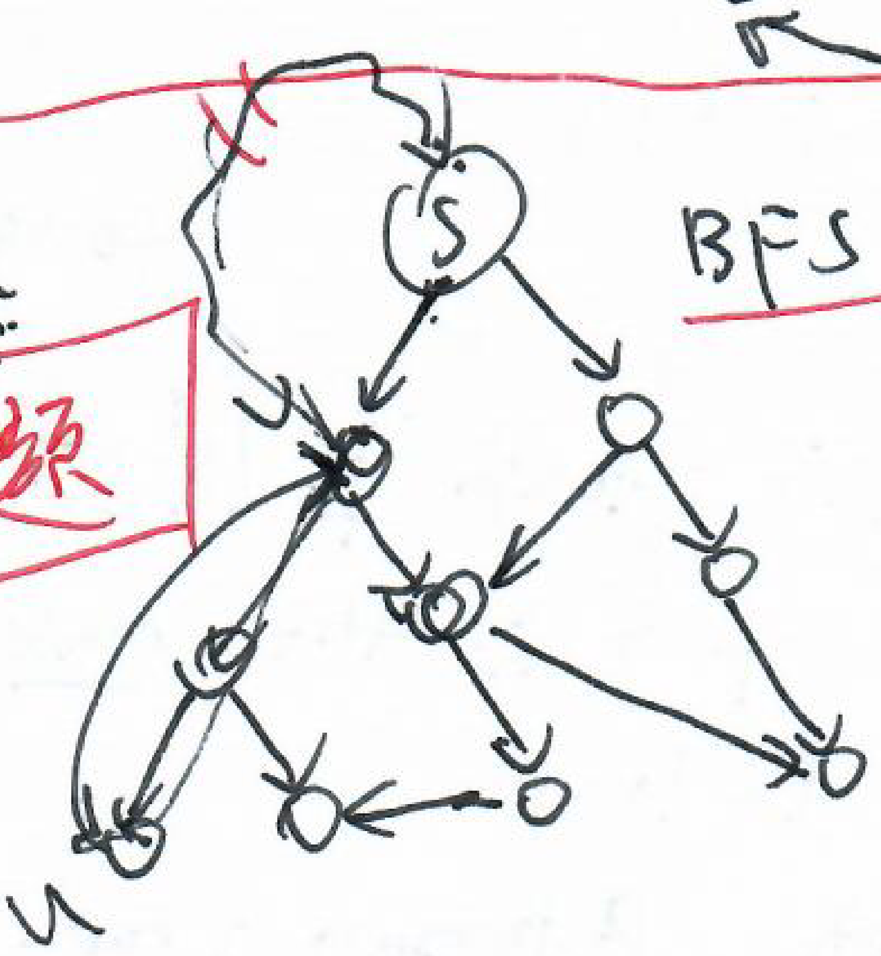
(2) "多余 v → w" in wrong direction. (parity 不变)

"w ⇝ v" (Path P) in scc.

(2.1) P is of odd length

(2.2) P is of even length.

$\boxed{\text{Alg}}$: ③ 思考题



BFS from s. (tree + cross + back)

Alg:

① 分离各个 scc

② ~~scc 各个~~ $\boxed{\text{BFS}}$ from s

levels

Pf: (1) the same parity "nb" ⇒ 有 ~~奇圈~~ 检测出来.

(2) not the same parity: