

学习笔记

更新:2021 年 1 月 29 日

目录

1	算法	4
1.1	Morris traversal	4
1.2	Brian Kernighan 算法	4
1.3	欧拉回路/通路	4
1.4	AVL tree	5
1.5	数位 dp	5
1.6	lca	5
1.7	差分树	5
1.8	并查集	5
1.9	矩阵树定理	6
1.10	滚动哈希 Rabin-Karp 算法	6
2	软件	7

2.1	git	7
3	something	8
3.1	#if #else #endif	8
4	tikz	8

1 算法

1.1 Morris traversal

迭代, $O(1)$ 空间, 修改叶结点的左右节点实现遍历。

中序:

根结点出发, 记录当前节点。重复以下过程至当前节点为空:

1、若当前节点无左结点, 输出并转移至右节点;

2、考虑当前节点的前驱节点, 无右节点则将其右节点指向当前节点且当前节点左移, 有右节点则当前节点变为其右节点, 再将该节点的右节点还原为空, 输出当前节点, 当前节点右移。

1.2 Brian Kernighan 算法

清除二进制最右边的 1 ($\text{number} - 1$ 和 number 与运算)

1.3 欧拉回路/通路

定义: 经过图中所有边一次且经过所有顶点的回路/通路

判定：

无向图 G 存在欧拉通路的充要条件是：

G 为连通图, 并且 G 仅有两个奇度结点(度数为奇数的顶点)或者无奇度结点。

有向图 D 存在欧拉通路的充要条件是：

D 为有向图, D 的基图连通, 并且所有顶点的出度与入度都相等; 或者除两个顶点外, 其余顶点的出度与入度都相等, 而这两个顶点中一个顶点的出度与入度之差为 1, 另一个顶点的出度与入度之差为 -1。

1.4 AVL tree

1.5 数位 dp

1.6 lca

1.7 差分树

1.8 并查集

按秩合并可撤销

1.9 矩阵树定理

1.10 滚动哈希 Rabin-Karp 算法

选取两个合适的互素常数 b 和 h ($b, h >$ 字符最大值)。设字符串 $C=c_1c_2\dots c_m$, 取哈希函数 $H(C) = (c_1b^{m-1} + c_2b^{m-2} + \dots + c_m) \bmod h$ 。(把字符串看作 b 进制数)

可以用 64 位无符号整数计算哈希值, 并取 h 等于 2^{64} , 省去求模运算。

2 软件

2.1 git

安装: `apt-get install git`

设置: `git config --global (user.name "", user.email"", core.editor vim, color.ui true)`

初始化: `git init`

查看存档及变化信息: `git log/status`

跟踪文件: `git add (filename/-A 全选)(可编辑.gitignore)`

存档: `git commit`

读档: `git reset --hard (hash code)(清除所有新的记录)`

分支: `git branch; git checkout (-B 修改输出至新分支) 分支名`

3 something

注意数据范围

上取整: $(a-1)/b+1$

3.1 #if #else #endif

c 语言字符串常量写一起会拼接

gcc -verbose

4 tikz

