Hw 9

Page297：19.2-1

Page300：19.3-1

Page302：19.4-2

Page327：21.2-2

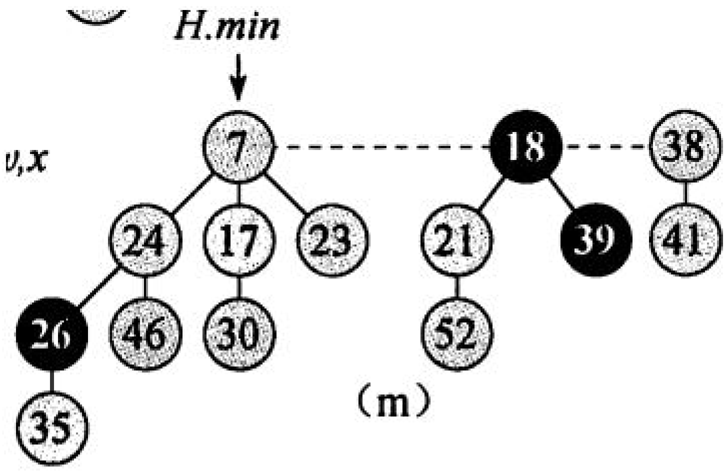
Page330：21.3-3

Page47：4.2-3

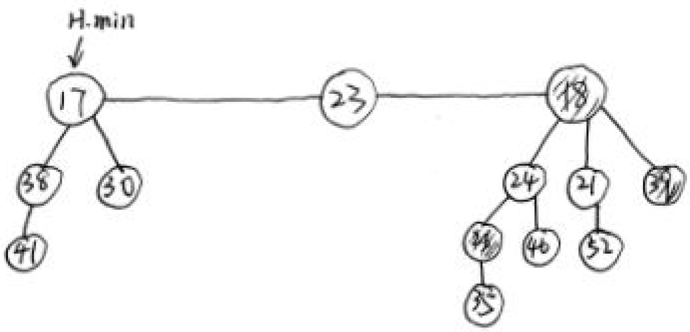
19.2

Page297：19.2-1

19.2-1 给出图 19-4(m)中的斐波那契堆调用 FIB-HEAP-EXTRACT-MIN 后得到的斐波那契堆。



(m)



19.3

Page300：19.3-1

19.3-1 假定斐波那契堆中一个根 x 被标记了。解释 x 是如何成为一个被标记的根的。试说明 x 是否被标记对分析并没有影响，即使它不是一个先被链接到另一个结点，后又丢失了一个孩子的根。

答：x先成为H.min指向的根节点的孩子，然后经FIB-HEAP-DELETE失去一个孩子，并被标记，然后H.min指向的根节点经FIB-EXTRACT-MIN被抽出，于是x成为一个被标记的根节点。此外，DECREASE-KEY不会导致被标记的根。

FIB-HEAP的所有操作均不对一个根节点的mark属性进行检测，因此x是否被标记对分析并无影响。

by 59

19.3-1 x先成为H.min指向的根节点的孩子，然后经FIB-HEAP-DELETE失去一个孩子，并被标记；然后H.min指向的根节点经FIB-EXTRACT-MIN被抽出，于是x成为一个被标记的根节点。

FIB-HEAP的所有操作均不对一个根节点的mark属性进行检测，因此x是否被标记对分析并无影响。

19.4

Page302：19.4-2

19.4-2 假定对级联切断操作进行推广，对于某个整数常数 k，只要一个结点失去了它的第 k 个孩子，就将其从它的父结点上剪切掉(19.3 节中为 $k=2$ 的情形)。k 取什么值时，有 $D(n)=O(\lg n)$。

by 59

19.4-2 对FIB-HEAP的任一节点x，假定$x.degree=m$，x的孩子为$y_1, y_2, \dots, y_m$，则由引理19.1，$y_1.degree \geq 0$，且对于$i=2,3,\dots,m$，$y_i.degree \geq i-2 \geq i-k$。

记S_k为x为根的子树中至少应有的结点数，则有$S_0=1, S_1=2, \dots, S_{k-1}=k$。

$S_m = k + \sum_{i=0}^{m-k}$，$S_m - S_{m-1} = S_{m-k}$。

$S_m = \begin{cases} m+1, & 0 \leq m < k \\ S_{m-1}+S_{m-k}, & m \geq k \end{cases}$，序列$\{S_m\}$的生成函数为$F(x)=\frac{1-x^k}{(1-x)(1-x-x^k)}$，

令α为方程$x^k=x^{k-1}+1$的根，有$f_k=S_k$，

$f_k=k+1 \geq 1=\alpha^0$，$f_{k+1}=k+3 \geq \alpha^1$，……，$f_{k+k}=k+\frac{(k+1)(k+2)}{2} \geq 2k+1+\alpha^{k-1} \geq \alpha^k$。

则有$f_{m+k}=f_{m+k-1}+f_m \geq \alpha^{m-1}+\alpha^{m-k}=\alpha^m$。

又$f_0=1$，$f_k=k+1=f_0+k$，$f_{m+k}=f_{m+k-1}+f_m=k+\sum_{i=0}^{m}f_i$，可推出$f_{m+k}=k+\sum_{i=0}^{m}f_i$。

对$\forall 0 \leq i \leq k$，$S_i \geq f_{i+k}$，则对于$m \geq k$有$S_m=k+\sum_{i=0}^{m-k}S_i \geq k+\sum_{i=0}^{m-k}f_{i+k}=k+\sum_{i=0}^{m}f_i=f_{m+k}$。

x为x为根的子树大小$size(x) \geq S_m \geq f_{m+k} \geq \alpha^m$，即$\log_\alpha n \geq m$。

则$k=O(\lg n)$时有$D(n)=O(\lg n)$。

即$k \geq 2$

21.2

Page327：21.2-2

21.2-2 给出下面程序的结果数据结构，并回答该程序中 FIND-SET 操作返回的答案。这里使用加权合并启发式策略的链表表示。

```
1  for i=1 to 16
2      MAKE-SET(x_i)
3  for i=1 to 15 by 2
4      UNION(x_i, x_{i+1})
5  for i=1 to 13 by 4
6      UNION(x_i, x_{i+2})
7  UNION(x_1, x_5)
8  UNION(x_11, x_13)
9  UNION(x_1, x_10)
10 FIND-SET(x_2)
11 FIND-SET(x_9)
```

假定如果包含 x_i 和 x_j 的集合有相同的大小，则 UNION(x_i，x_j)表示将 x_j 所在的表链接到 x_i 所在的表后。

答：

程序的结果数据结构是链表。在加权合并启发式策略下，MAKE-SET(x)、UNION(xi, xj)、FIND-SET(x)操作解释如下。

MAKE-SET(x): 创建一个链表，仅有一个节点x。

UNION(xi, xj): 将xi与xj进行拼接，由于这里采用加权合并启发式策略，则表示将较短的表拼接到较长的表中。

FIND-SET(x): 返回一个指针，这个指针指向包含x的（唯一）集合的代表。

1~2行执行结果：x1, x2, ..., x16

3~4行执行结果：x1 = (x1, x2)

x3 = (x3, x4)

……

x15 = (x15, x16)

5~6行执行结果：x1 = (x1, x2, x3, x4)

x5 = (x5, x6, x7, x8)

x9 = (x9, x10, x11, x12)

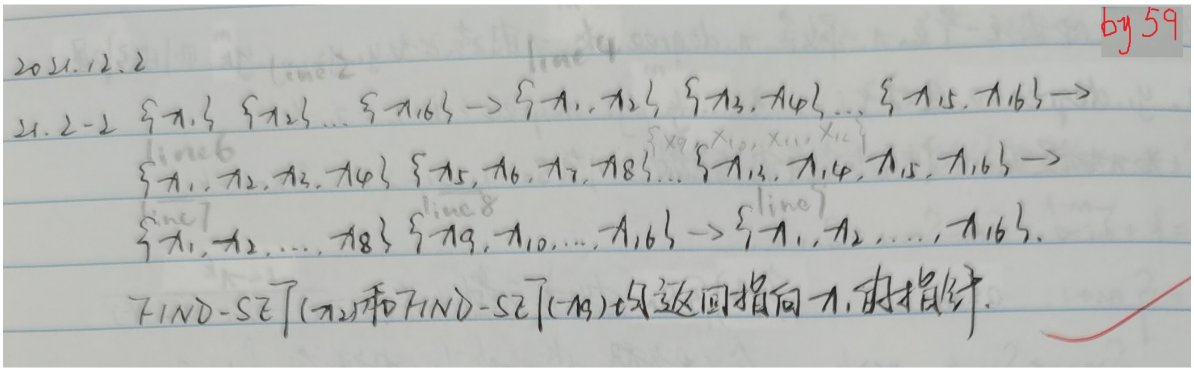
x13 = (x13, x14, x15, x16)

第7行执行结果：x1 = (x1, x2, x3, x4, x5, x6, x7, x8)

第8行执行结果：x13 = (x9, x10, x11, x12, x13, x14, x15, x16)

第9行执行结果：x1 = (x1, x2, x3,……, x15, x16)

第10~11行执行结果：均是返回指向x1的指针


by 59

2021.12.2

21.2-2 $\{x_1\}\ \{x_2\} \dots \{x_{16}\} \to \{x_1, x_2\}\ \{x_3, x_4\} \dots \{x_{15}, x_{16}\} \to$

$\{x_1, x_2, x_3, x_4\}\ \{x_5, x_6, x_7, x_8\} \dots \{x_{13}, x_{14}, x_{15}, x_{16}\} \to$

$\{x_1, x_2, \dots, x_8\}\ \{x_9, x_{10}, \dots, x_{16}\} \to \{x_1, x_2, \dots, x_{16}\}$.

FIND-SET(x_2)和FIND-SET(x_9)均返回指向x_1的指针.

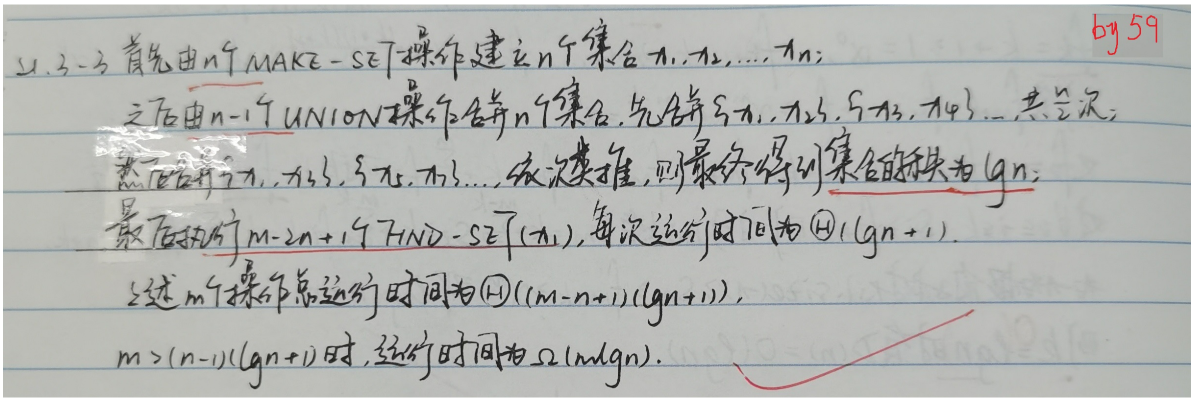
21.3

Page330：21.3-3

21.3-3 给出一个包含 m 个 MAKE-SET、UNION 和 FIND-SET 操作的序列（其中有 n 个是 MAKE-SET 操作），当仅使用按秩合并时，需要 $\Omega(m\lg n)$ 的时间。

答:

首先，进行n次MAKE-SET(x)操作创建单节点集合{x1}, {x2},...,{xn}。之后，进行n-1 UNION(xi, xj)操作创建树深为lgn的集合。最后，进行m-2n+1次FIND-SET(x)操作。每个FIND-SET(x)需要Ω(lgn)的时间。让m≥3n，则需要超过m/3次FIND-SET(x)操作，因此需要Ω(mlgn)的时间。


by 59

21.3-3 首先由n个MAKE-SET操作建立n个集合 $x_1, x_2, \dots, x_n$;

之后由n-1个UNION操作合并n个集合, 先合并$\{x_1, x_2\}, \{x_3, x_4\}\dots$ 共$\frac{n}{2}$次;

然后合并$\{x_1, \dots, x_4\}, \{x_5, \dots, x_8\}\dots$, 依次类推, 则最终得到集合的秩为$\lg n$;

最后执行m-2n+1个FIND-SET(x_1), 每次运行时间为$\Theta(\lg n+1)$.

上述m个操作总运行时间为$\Theta((m-n+1)(\lg n+1))$,

$m>(n-1)(\lg n+1)$时, 运行时间为$\Omega(m\lg n)$.

4.2

Page47：4.2-3

4.2-3 如何修改 Strassen 算法，使之适应矩阵规模 n 不是 2 的幂的情况？证明：算法的运行时间为 $\Theta(n^{\lg 7})$。

原Strassen算法复杂度就是$\Theta(n^{\lg 7})$。当矩阵规模n不是2的幂时，可用0填充，将矩阵规模扩成2的幂，复杂度不变。严格证明：

用0补足行列，使之变为规模为2的幂（m）的情况，记n<m<2n，则

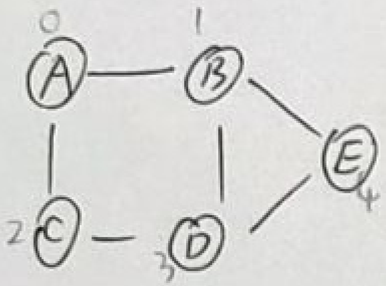
$$T(n) \le c_1(m^{lg7}) \le c_1(2n^{lg7}) \le c_1 * 2lg7 * n^{lg7}$$
$$T(n) \ge c_2(m^{lg7}) \ge c_2 n^{lg7}$$
$$\text{综上，} T(n) = \Theta(n^{lg7})$$

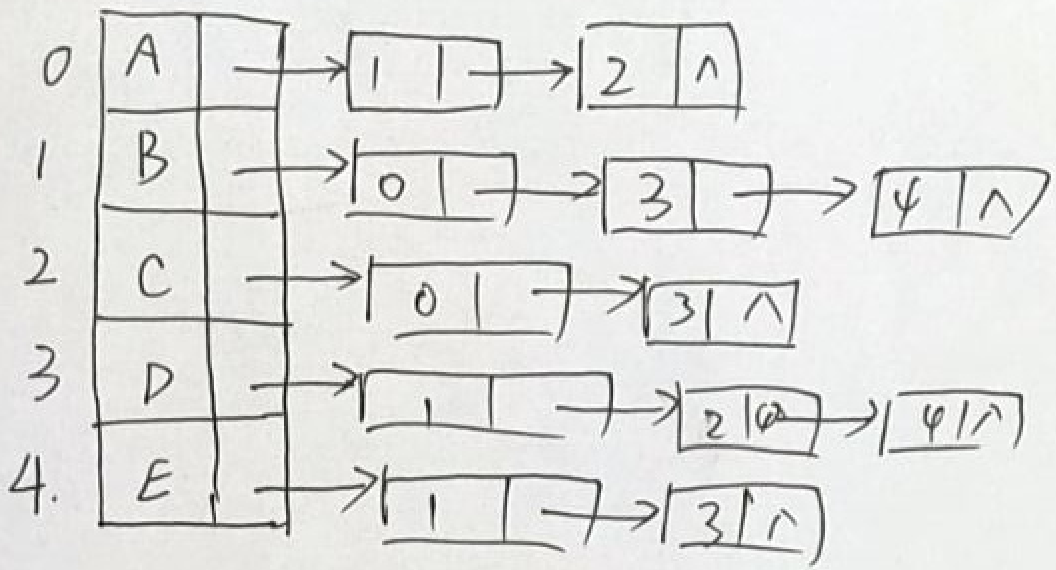
第五次随测答案

题目：画出所给图的邻接表。

答：主要注意邻接链表不能直接按照A、B、C等这种存，需要编号后存序号。

参考#65同学


0
1
A
B
E
4
2
C
3
D


0 A → 1 → 2 ∧
1 B → 0 → 3 → 4 ∧
2 C → 0 → 3 ∧
3 D → 1 → 2 → 4 ∧
4. E → 1 → 3 ∧