装订线 答题时不要超过此线

中国科学技术大学

2017—2018学年 第 1 学期考试试卷

考试科目: 算法基础　　　　得分:__________

学生所在系:__________ 姓名:__________ 学号:__________

一、基本题: （共20分，每小题 5 分）

1. 请证明 $2n^2 + 5n \ \log n = O(n^3)$ （给出证明过程）
2. 请问关键字序列: **65，78，113，94，91，87，89** 是否是某棵二分检索树的一个合理检索序列? 合理 （请说明理由）
3. 求解递归方程 $T(n) = 4T(n/2+11) + n - 22$ （给出推导过程）
4. 求解递归方程 $T(n) = T(n/2) + T(n/3) + 2n$ （给出推导过程）

二、计算题: （共 40 分，每题10分）

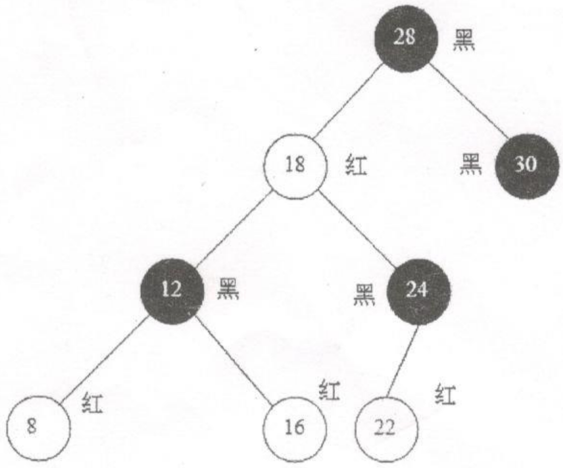
1. 已知有如下的函数 $G(A,n)$，分析该算法的最坏情况时间复杂度（表示成 n 的函数），初始调用时数组 $A[1..12] = (14, 32, 45, 64, 23, 68, 30, 49, 52, 21, 28, 62)$，请给出算法运行返回的最终结果，给出计算过程。

```
FUNCTION G(A, n)
/* A is an array of size n. */
if (n = 1) then   return A[1];
else
  for i = 1 to ⌊n/2⌋ do
     B[i] = max{A[2i], A[2i − 1]};
  end for
  x = G(B, ⌊n/2⌋);
  x = max{x, A[n]};
return (x)
```

2. 现在要求出 6 个矩阵的链乘 $A_1 \times A_2 \times A_3 ... \times A_6$，其中 A_i 是一个 $P_{i-1} \times P_i$ 的矩阵，$P[0..6] = (5, 10, 10, 40, 20, 10, 5)$ 请计算其最优的乘法次序，给出计算过程。

3. 已知模式串 $P[1..15] = (aaabcaaaabcaaab)$，请根据教材中 KMP 算法，给出数组 $\pi[1..15]$ 的值，给出计算过程。

4. 已知有如下所示的红黑树，请给出在这棵红黑树中**依次**执行下列操作的**过程和结果**：① 插入 10，② 插入 20 ③删除 22



三、问答题：　　　　（共 20 分，每小题 10 分）

1. 已知 H_1 和 H_2 分别是有 n_1 个 n_2 结点斐波那契堆，请问合并这两个斐波那契堆的真实代价是多少？如果在 Dijkstra 单源最短路径算法中优先队列选用斐波那契堆，则此时 Dijkstra 单源最短路径算法的时间复杂度是多少？给出结论和说明，不用写算法。

2. 设有大小不同的 n 个瓶塞和 n 个瓶子，但由于它们之间的差异很小，无法凭眼睛分辨出它们的大小，只能通过试探看看瓶塞和瓶子是否匹配（假设一次试探用一个单位时间），请问你有没有办法，仅用 $\Theta(n\log n)$ 的**期望**时间，把这 n 个瓶塞和 n 个瓶子完全匹配上。（只需说明你的方法和简单的分析，无需写算法）

四、算法设计：　　　　（共 20 分，每题 10 分）

1. 已知数组 $A[1..n]$ 中元素两两不同，数组 $B[1..n]$ 中的元素也两两互异，A 和 B 中元素均属于某个有序集。请设计一个最坏时间复杂度为 $O(n\log n)$ 的算法，找出 $A[1..n]$ 和 $B[1..n]$ 中所有的相同元素。

2. 设 G 是一个无向连通图，图中边有非负权值，请设计算法判断图 G 中是否存在回路，如果存在回路，则求出图 G 中的最小权值回路（回路中所有边的权值之和最小），要求算法的最坏时间复杂度为 $O(|V|^3)$。

— 完 —

2017–2018 学年 第 1 学期考试试卷

一、基本题

1.证明：

当$n \geq 1$时，总有$n > \log n$成立。

故$2n^2 + 5n\log n$

$\leq 2n^2 + 5n^2$

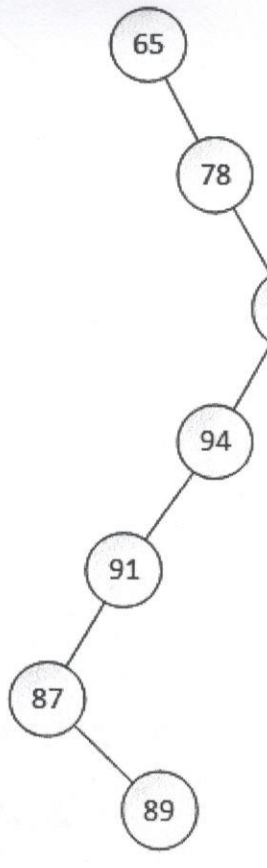
$= 7n^2$

$\leq 7n^3$

故存在正常数$c = 7$和$n_0 = 1$，使得对所有$n \geq 1$，都有$2n^2 + 5n\log n \leq 7n^3$成立。

因此，$2n^2 + 5n\log n = O\left(n^3\right)$得证。

2.答：如图所示，该关键字序列是一棵二分检索树的一个合理检索序列。



3.解：

令 $n^{'}=n-22$，则 $n=n^{'}+22$

故原式可转化为 $T\left(n^{'}+22\right)=4T\left(\frac{n^{'}}{2}+22\right)+n^{'}$。

令 $F\left(n\right)=T\left(n^{'}+22\right)$

$$F\left(n\right)=4F\left(\frac{n}{2}\right)+n$$

运用主方法，$a=4,b=2,f\left(n\right)=n$

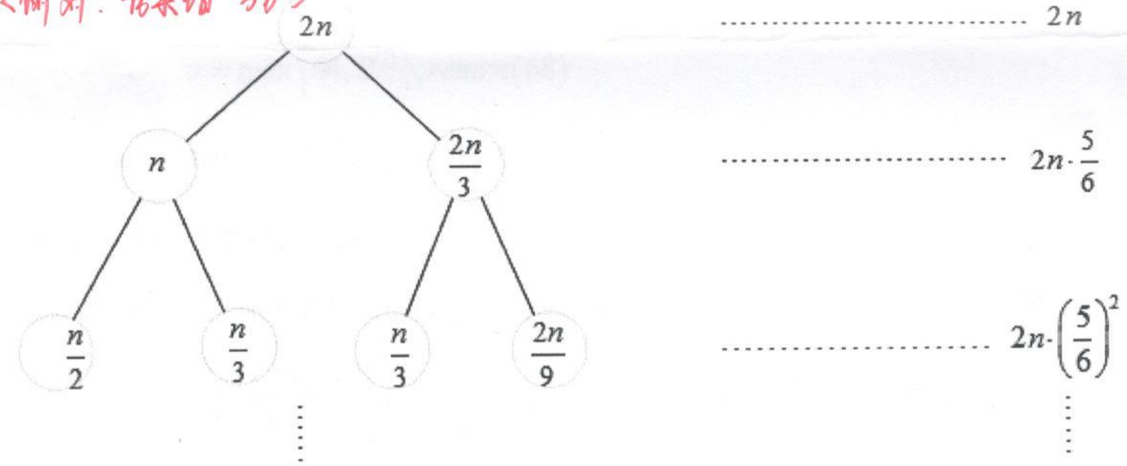
故 $f\left(n\right)=O\left(n^{\log_b a-\varepsilon}\right)=O\left(n^{\log_2 4-\varepsilon}\right)$（取 $\varepsilon=2$）

$\therefore T\left(n\right)=\Theta\left(n^{\log_b a}\right)=\Theta\left(n^2\right)$　　〈结果对. 过程错. 2分〉

4.解：

其递归树如下所示，

〈树对. 结果错 3分〉



从第 $\lfloor\log_3 n\rfloor$ 层以后，递归树最右边分支结束，不再计算下面层数，有

$$T\left(n\right)\geq 2n\left(1+\frac{5}{6}+\left(\frac{5}{6}\right)^2+\cdots+\left(\frac{5}{6}\right)^{\lfloor\log_3 n\rfloor}\right)=2n\frac{\left(1-\left(\frac{5}{6}\right)^{\lfloor\log_3 n\rfloor+1}\right)}{1-\frac{5}{6}}\geq 2n=o\left(n\right)$$

在第 $\lfloor\log_2 n\rfloor$ 层以后，递归树最左边的分支结束。又因为在第 $\lfloor\log_2 n\rfloor$ 层以后，递归树，但按满递归树计算，故

$$T(n)\le 2n\left(1+\frac{5}{6}+\cdots+\left(\frac{5}{6}\right)^{\lfloor\log_2 n\rfloor}\right)=2n\frac{\left(1-\left(\frac{5}{6}\right)^{\lfloor\log_2 n\rfloor+1}\right)}{1-\frac{5}{6}}\le 12n=O(n)$$

故$T(n)=\Theta(n)$。

二、计算题

1.解：

计算过程：

$G(A,12)$

$\quad G\left(B^{(0)},6\right),B^{(0)}[1\cdots 6]=(32,64,68,49,52,62)$

$\quad\quad G\left(B^{(1)},3\right),B^{(1)}[1\cdots 3]=(64,68,62)$

$\quad\quad\quad G\left(B^{(2)},1\right),B^{(2)}[1]=(64),return(64)$，$x=64$

$\quad\quad\quad x=\max\left\{64,B^{(1)}\right\},return(68)$

$\quad\quad x=\max\left\{68,B^{(0)}\right\},return(68)$

$\quad x=\max\left\{68,A\right\},return(68)$

最坏情况下，for 循环内的语句执行次数为

$$\left\lfloor\frac{n}{2}\right\rfloor+\left\lfloor\frac{1}{2}\left\lfloor\frac{n}{2}\right\rfloor\right\rfloor+\left\lfloor\frac{1}{2}\left\lfloor\frac{1}{2}\left\lfloor\frac{n}{2}\right\rfloor\right\rfloor\right\rfloor+\cdots+1\le\frac{n}{2}+\frac{n}{4}+\frac{n}{8}+\cdots+1\le n$$

而x一共比较了$\log_2{}^n$次，故最坏情况时间复杂度为$\Theta\left(n+\log_2{}^n\right)=\Theta(n)$。 (2) 缺(错)时间复杂度，扣 2分

2.解：A1 5*10　　A2 10*10　　A3 10*40　　A4 40*20　　A5 20*10　　A6 10*5

先计算后画图，计算过程如下：

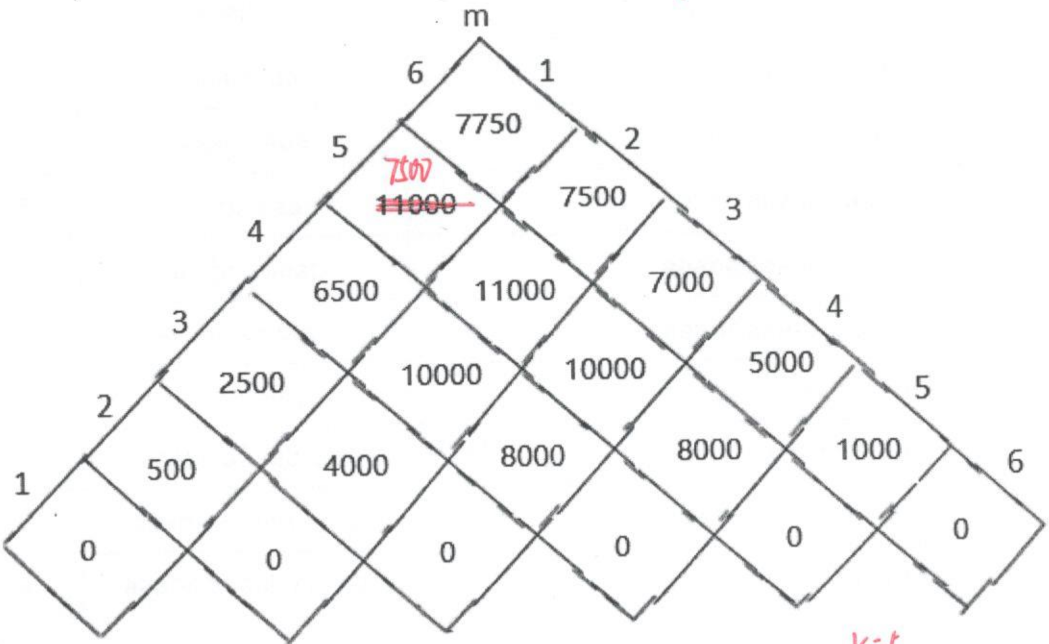
m[1,2]=m[1,1]+m[2,2]+p0p1p2 = 5*10*10 = 500　　k=1
m[2,3]=m[2,2]+m[3,3]+p1p2p3 = 10*10*40 = 4000　　k=2
m[3,4]=m[3,3]+m[4,4]+p2p3p4 = 10*40*20 = 8000　　k=3
m[4,5]=m[4,4]+m[5,5]+p3p4p5 = 40*20*10 = 8000　　k=4
m[5,6]=m[5,5]+ m[6,6]+ p4p5p6 = 20*10*5 = 1000　　k=5

m[1,3]=min(
　　m[1,1]+m[2,3]+p0p1p3 = 4000+5*10*40 = 6000　k=1
　　m[1,2]+m[3,3]+p0p2p3 = 500+5*10*40 = 2500　k=2
)

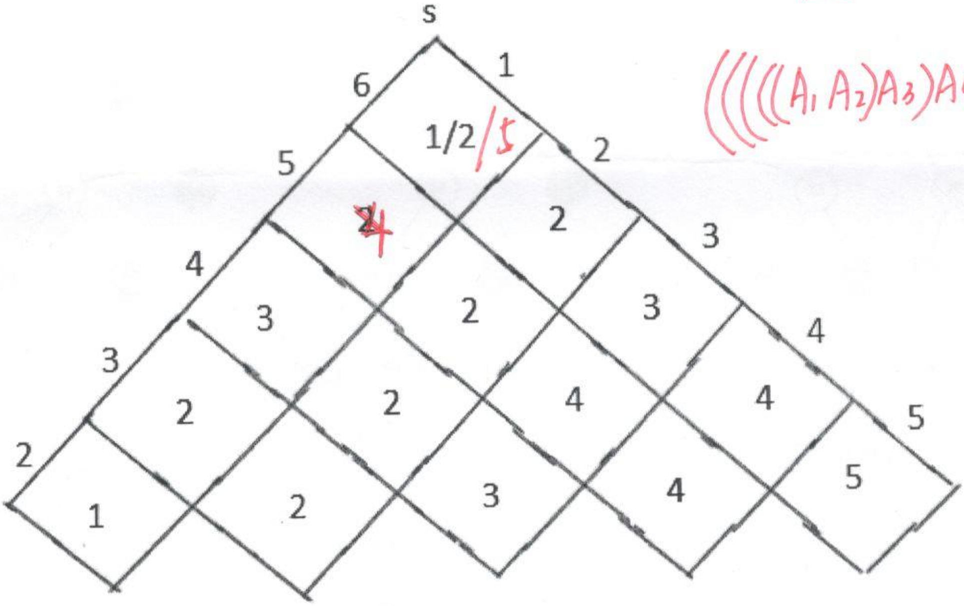
m[2,4]=min(
m[2,2]+m[3,4]+p1p2p4 = 8000+10*10*20 = 10000　　k=2
m[2,3]+m[4,4]+p1p3p4 = 4000+10*40*20 = 12000　　k=3
)
m[3,5]=min(
m[3,3]+m[4,5]+p2p3p5 = 8000+10*40*10 = 12000　　k=3
m[3,4]+m[5,5]+p2p4p5 = 8000+10*20*10 = 10000　　k=4
)
m[4,6]=min(
m[4,4]+m[5,6]+p3p4p6 = 1000+40*20*5 = 5000　　k=4
m[4,5]+m[6,6]+p3p5p6 = 8000+40*10*5 = 10000　　k=5
)
m[1,4]=min(
m[1,1]+m[2,4]+p0p1p4 = 10000+5*10*20 = 11000　　k=1
m[1,2]+m[3,4]+p0p2p4 = 500+8000+5*10*20 = 9500　　k=2
m[1,3]+m[4,4]+p0p3p4 = 2500+5*40*20 = 6500　　k=3
)
m[2,5]=min(
m[2,2]+m[3,5]+p1p2p5 = 10000+10*10*10 = 11000　　k=2
m[2,3]+m[4,5]+p1p3p5 = 4000+8000+10*40*10 = 16000　　k=3
m[2,4]+m[5,5]+p1p4p5 = 10000+10*20*10 = 12000　　k=4
)
m[3,6]=min(
m[3,3]+m[4,6]+p2p3p6 = 5000+10*40*5 = 7000　　k=3
m[3,4]+m[5,6]+p2p4p6 = 8000+1000+10*20*5 = 10000　　k=4
m[3,5]+m[6,6]+p2p5p6 = 10000+10*10*5 = 10500　　k=5
)
m[1,5]=min(
m[1,1]+m[2,5]+p0p1p5 = 11000+5*10*10 = 11500　　k=1
m[1,2]+m[3,5]+p0p2p5 =500+10000+5*10*10 = 11000　　k=2
m[1,3]+m[4,5]+p0p3p5 = 2500+8000+5*40*10 = 12500　　k=3
m[1,4]+m[5,5]+p0p4p5 = 6500+~~8000~~+5*20*10 = ~~15500~~　　k=4
) 6500　　7500
m[2,6]=min(
m[2,2]+m[3,6]+p1p2p6 = 7000+10*10*5 = 7500　　k=2
m[2,3]+m[4,6]+p1p3p6 = 4000+5000+10*40*5 = 11000　　k=3
m[2,4]+m[5,6]+p1p4p6 = 10000+1000+10*20*5 = 12000　　k=4
m[2,5]+m[6,6]+p1p5p6 = 11000+10*10*5 = 11500　　k=5
)
m[1,6]=min(
m[1,1]+m[2,6]+p0p1p6 = 7500+5*10*5 = 7750　　k=1
m[1,2]+m[3,6]+p0p2p6 = 500+7000+5*10*5 = 7750　　k=2
m[1,3]+m[4,6]+p0p3p6 = 2500+5000+5*40*5 = 8500　　k=3
m[1,4]+m[5,6]+p0p4p6 = 6500+1000+5*20*5 = 8000　　k=4

m[1,5]+m[6,6]+p0p5p6 = ~~11000~~ 7500 +5*10*5 = ~~11250~~ 7750　　　k=5

)



K=5



(((((A1 A2)A3)A4)A5) A6)

所以，最优乘法次序应该为(A1(A2(A3(A4(A5A6)))))或((A1A2)(A3(A4(A5A6))))

3.解：　〈无过程 -2分〉

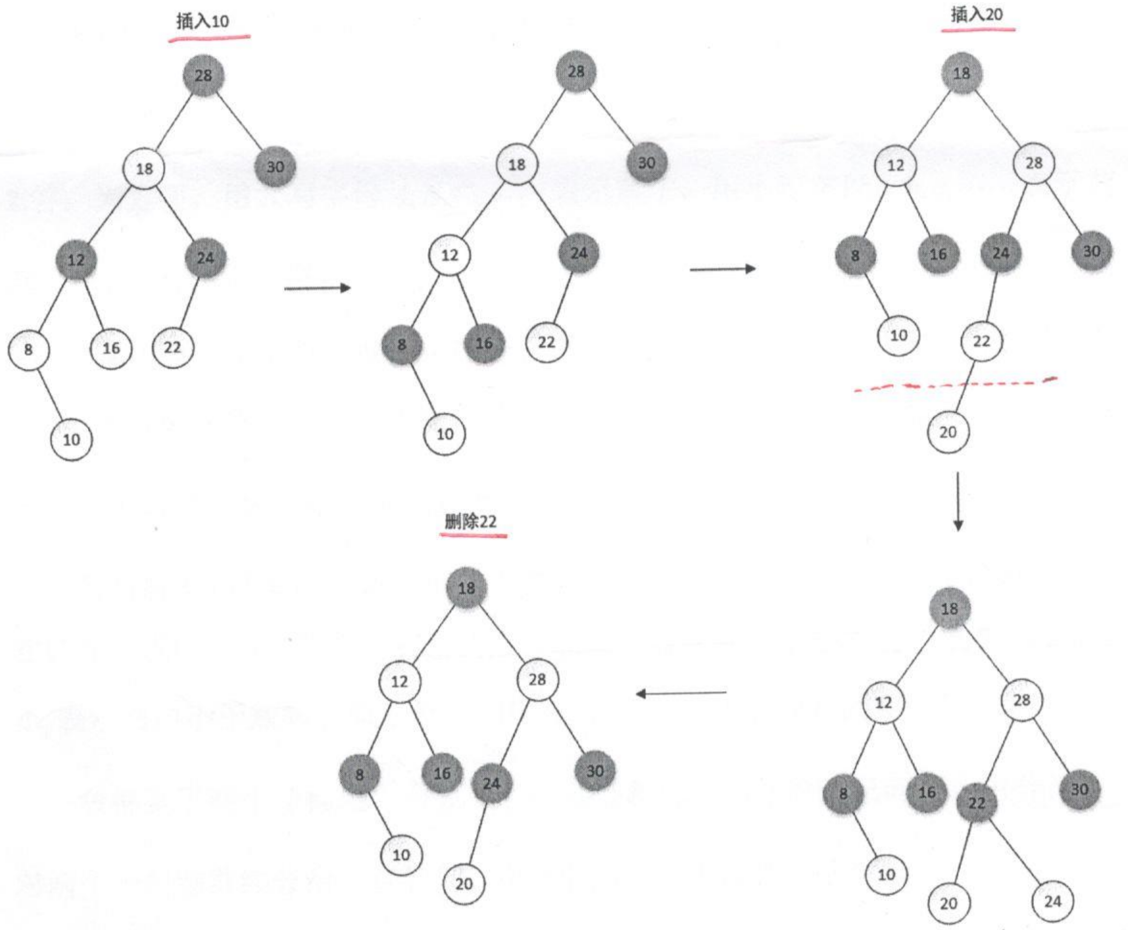
i	$\pi[i]$	理由	
1	0	去掉第一个字符，剩余部分是空字符 ε，ε 作为后缀不与 P 的非空前缀匹配	
2	1	aa	aaabc aaaab caaab
3	2	aaa	aaabc aaaab caaab
4	0	aaab	aaabc aaaab caaab

K=1　　　　　　　　　　K=2

(A1(A2(A3(A4(A5 A6)))))　　　　((A1 A2)(A3(A4(A5 A6))))

5	0	aaabc	aaabc aaaab caaab
6	1	aaabc a	aaabc aaaab caaab
7	2	aaabc aa	aaabc aaaab caaab
8	3	aaabc aaa	aaabc aaaab caaab
9	3	aaabc aaaa	aaabc aaaab caaab
10	4	aaabc aaaab	aaabc aaaab caaab
11	5	aaabc aaaab c	aaabc aaaab caaab
12	6	aaabc aaaab ca	aaabc aaaab caaab
13	7	aaabc aaaab caa	aaabc aaaab caaab
14	8	aaabc aaaab caaa	aaabc aaaab caaab
15	4	aaabc aaaab caaab	aaabc aaaab caaab

4.解：



三、问答题

1.答：合并结点数分别为n_1和n_2的斐波那契堆H_1和H_2，真实代价为$O(1)$。

$\because$斐波那契堆的势函数为$\Phi(H)=t(H)+2m(H)$，故合并后的势函数变化为

$$\Phi(H)-\left(\Phi(H_1)+\Phi(H_2)\right)=\left(t(H)+2m(H)\right)-\left(\left(t(H_1)+2m(H_1)\right)+\right)\left(t(H_2)+2m(H_2)\right)$$
$$=\left(t(H)-\left(t(H_1)+t(H_2)\right)\right)+2\left(m(H)-\left(m(H_1)+m(H_2)\right)\right)$$

又$\because$ $t(H)=t(H_1)+t(H_2)$和$m(H)=\mathrm{m}(H_1)+m(H_2)$

故$\Phi(H)-\left(\Phi(H_1)+\Phi(H_2)\right)=0$，即合并这两个斐波那契堆的真实代价为$O(1)$。 4分

在 Dijkstra 单源最短路径算法中优先队列选用斐波那契堆，则时间复杂度为$O(|V|\lg|V|+|E|)$。因为每次 EXTRACT-MIN 操作的摊还代价为$O(\lg|V|)$，每次 DECREASE-KEY 操作的摊还代价为$O(1)$，而一共执行了$|V|$次 EXTRACT-MIN 操作和$|E|$次 DECREASE-KEY 操作，故时间复杂度为$O(|V|\lg|V|+|E|)$。

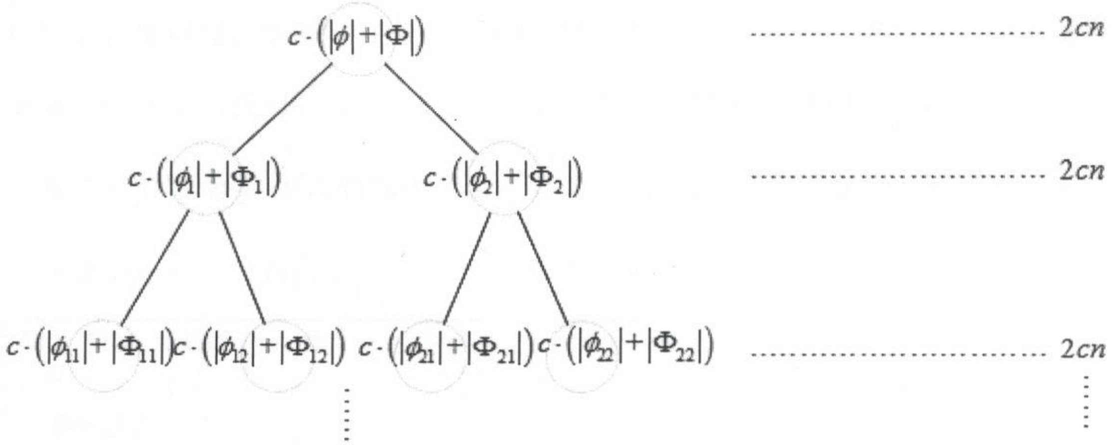
2.答：为方便，用大写字母$A,B,C,\cdots\cdots$表示瓶子，用小写字母$a,b,c,\cdots\cdots$表示瓶塞，用(a,A)表示瓶塞a和瓶子A匹配。

从n个瓶子中随机选取一个瓶子I,逐个把瓶塞与瓶子进行匹配，此过程耗时为n。在匹配过程中，将大于瓶口的塞和小于瓶口的塞分成两个集合，其中用ϕ_1表示大于瓶口的塞子集合和用ϕ_2表示小于瓶口的瓶塞集合。 瓶塞 分为2部分

用与瓶子I匹配的瓶塞i，逐个与剩余的瓶子进行匹配，此过程耗时为$n-1$。在匹配过程中，将瓶口小于瓶塞的瓶子和大于瓶塞的瓶子分成两个集合，其中用Φ_1表示瓶口小于瓶塞的瓶子集合和用Φ_2表示瓶口大于瓶塞的瓶子集合。 瓶子 分为2部分

故得到了两个子问题，分别是$(\phi_1,\ \Phi_1)$和$(\phi_2,\ \Phi_2)$的匹配问题。由分治法，对两个子问题继续分治，直到$(\phi_1,\ \Phi_1)$和$(\phi_2,\ \Phi_2)$的规模都为 1。

整个算法类似 QUICK-SORT，由于$(\phi_1,\ \Phi_1)$和$(\phi_2,\ \Phi_2)$在各层分治匹配过程中，只被比较一次，故递归树为



其树高为$\log n$，故其期望时间为$\Theta(n\log n)$。

四、算法设计

1. 答：首先使用最坏时间复杂度不超过$O(n\log n)$的排序算法如堆排序，对数组A和B进行排序，此过程在最坏情况下耗时为$O(n\log n)$。其次采用类似归并操作的思想，扫描一遍数组A和B，找出所有相同的元素并放入到数组C中，此过程耗时为$O(n)$。故最坏情况下，总过程的时间复杂度为$O(n\log n+n)=O(n\log n)$。

排序错 -4

其伪代码如下所示，

```
FINDSAMEELEMENT(A, B, n)
   HEAPSORT(A)
   HEAPSORT(B)
   C = []
   i = 1, j = 1
   while i <= n and j <= n
      if A[i] < B[j]
         i++
      else if A[i] == B[i]
         C.push(A[i])
         i++, j++
      else
         j++
      endif
   endwhile
   return C
```

2. 答：在 Floyd-Warshall 算法基础上进行修改，在更新最短路径前增加一个操作，即原来的路径与用新的节点为中间节点找出来的路径。若这两条路径不同，则把

DFS 给4分

他们的长度相加，得到的值为这两条路径组成的回路的权值，算法最后输出求得的最小的回路的权值。Floyd-Warshall 算法的时间复杂度为 $\Theta(|V|^3)$，修改后第三重循环中只增加了 $\Theta(1)$ 的操作，所以总共增加了 $\Theta(|V|^3)$ 的操作，所以算法最坏时间复杂度不超过 $\Theta(|V|^3)$。其伪代码如下所示：

```
Min_loop_weight(w, n)
//输入 w 为图的邻接矩阵，n 为点个数
   Weight = ∞
   for k = 1 to n
      for i = 1 to n
         for j = 1 to n
            if k != i and k != j and i != j
                Weight = min(Weight, w[i][j] + w[i][k] + w[k][j])
                w[i][j] = min( w[i][j], w[i][k] + w[k][j] )
            endif
         endfor
      endfor
   endfor
   return Weight
```