一、选择题（5 题，每题 2 分，共 10 分）

1、以下关于渐进符号性质表达正确的是（ C ）。

A、$O(f(n)+g(n))=O(\min\{f(n), g(n)\})$

B、$f(n)=O(g(n)), g(n)=O(h(n))\Rightarrow h(n)=O(f(n))$

C、$f(n)=\Theta(g(n)), g(n)=\Theta(h(n))\Rightarrow f(n)=\Theta(h(n))$

D、$f(n)=O(g(n))\Leftrightarrow g(n)=O(f(n))$

2、回溯法一般是按（ ）策略搜索问题的解空间树。

A、深度优先　B、广度优先　C、广度深度结合　D、活节点优先

3、适合应用贪心算法与动态规划算法求解的问题所具有的共同特点是（ ）。

A、重叠子问题　B、最优子结构性质　C、贪心选择性质　D、独立子问题

4、以下算法思想中，有可能得不到解，但不会得到不正确解的是（ ）。

A、贪心　B、舍伍德　C、蒙特卡罗　D、拉斯维加斯

5、对于物品数量为 n，背包容量为 c 的情况下的 0-1 背包问题，若用回溯法求解，算法的时间复杂度为（ D ）。

A、nc　B、2^{nc}　C、$c2^n$　D、$n2^n$

二、填空题（15 空，每空 2 分，共 30 分）

1、算法是由若干条指令组成的有穷序列，需要满足输入、输出、确定性、__(1)__和有限性。

2、算法的效率一般可以从两个方面衡量，分别称为__(2)__和空间复杂性。

3、假设算法 A 理论上的时间复杂度 $T(n)=2^n$，现在有两台同类型计算机 M1 和 M2，M2 的计算速度是 M1 的 64 倍。若在 M1 和 M2 上分别测试算法 A，则在相同时间内，M1 和 M2 能够求解的问题规模 n1 和 n2 的关系为__(3)__。若算法 B 理论上的时间复杂度为 $T(n)=n^2$，若在 M1 和 M2 上分别测试算法 B，则 M1 和 M2 求解相同规模问题所耗费的时间 t1 和 t2 的关系为__(4)__。

4、用回溯法求解问题时，可将问题解空间看作一棵树状结构，根据问题不同，解空间树一般可分为子集树和排列树两种结构，对于 N 后问题，其解空间树是__(5)__结构，对于旅行售货员 TSP 问题，其解空间树是__(6)__结构。

5、用分治法求解 n 个元素全排列问题时，算法时间复杂度递推方程为__(7)__。

6、0-1 背包问题可以使用动态规划、回溯和分支限界三种方法求解，其中排序对求解过程没有影响的方法是__(8)__。

7、若序列 X={A, B, C, B, D, A, B}，Y={B, D, C, A, B, A}，则 X 和 Y 的一个最长公共子序列为__(9)__；采用动态规划方法求解该类问题（序列 X 和 Y 的规模分别为 m 和 n）的时间复杂度为__(10)__。

8、采用分治法解决问题时，算法时间复杂度分析典型的递推方程为

$$T(n)=\begin{cases}O(1) & n=1\\ aT(n/b)+f(n) & n>1\end{cases}$$

，对于某一问题，若 $f(n)=O(1)$且 $a=1$，则求解该问题的时间复杂度为__(11)__；若 $f(n)=O(n)$且 $a=b>1$，则求解该问题的时间复杂度为__(12)__；若 $f(n)=O(n)$且 $a=b^2>1$，则求解该问题的时间复杂度为__(13)__；若已知 $f(n)=O(n)$，且 a，b 均大于 1，若希望将算法时间复杂度降低为不超过线性情况，则 a 和 b 应满足的条件为：__(14)__。

9、应用贪心算法求解问题的关键在于__(15)__。

三、算法分析题（3 题，每题 10 分，共 30 分）

1、设 A 为 n 个元素的数组，分析下面的算法并回答问题。（去年原题 3-1）

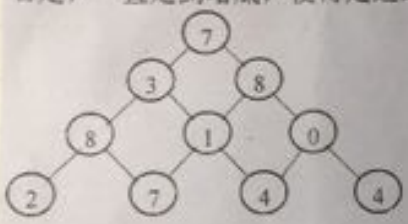
```
ALG(A[1..n])
  IF n=1 THEN RETURN A[n]
  ELSE temp=ALG(A[1..n-1])
    IF temp<A[n]
    THEN RETURN temp
    ELSE RETURN A[n]
```

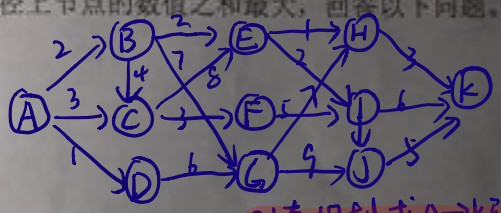
（1）该算法的输出是什么？

（2）以比较作为基本操作，给出算法 ALG 的时间复杂度递推方程 $T(n)$，并求解算法 ALG 的时间复杂度

（3）对于求解该类问题，算法 ALG 效率是否已最高，为什么？

2、有形如下图所示的数塔，若要求从数塔顶层出发，每个结点可以选择向左走或向右走，一直走到塔底，使得走过路径上节点的数值之和最大，回答以下问题。





动态规划 求A→K的数值和最小

（1）若用动态规划算法求解该问题，请给出最优值的递推定义，包括初值说明。

（2）针对上图的数塔，按照你所给出的递推定义，逐步求解从塔顶走到塔底的最大数值之和与得到最大数值和的路径。

拆：GDP最接近

3、已知有 n 个学生，若要求找出其中体重最为接近的两位学生，回答以下问题。

（1）用数学方法描述该问题求解目标；

（2）为该问题设计一个有效算法，需要给出算法思想和伪代码；

（3）对你所设计的算法进行时间复杂度分析。

四、算法设计与实现（2 题，每题 15 分，共 30 分）

1、设某游艇俱乐部在长江沿岸设置了 n 个游艇出租站，1, 2, …, n。游客可在这些游艇出租站租用游艇，并在任何另一个游艇出租站归还游艇。已知游艇出租站 i 到游艇出租站 j 之间的租金为 $r(i,j)$，$1 \le i < j \le n$。若要计算从游艇出租站 1 到游艇出租站 n 所需的最少花费，完成以下问题。

（1）若用动态规划方法求解该问题，给出问题解的递推定义；

（2）填空完成以下动态规划算法程序。

```
/**
 *@param n 游艇租赁站数量
 *@param r 租赁站之间的租赁费用矩阵
 */
int minRent(int n, int** r) {
    for(int k = 2; k < n; k++){
        for(int i = 0; ____(1)____; i++){
            int j = i+k;
            for(int p = i+1; ____(2)____; p++){
                int tmp = r[i][p] + r[p][j];
                if( ____(3)____ ) r[i][j] = tmp;
            }
        }
    }
    return ____(4)____;
}
```

2、羽毛球队有男女运动员各 n 人，给定两个 n 阶矩阵 P 和 Q，P[i][j]表示男运动员 i 和女运动员 j 配对组成混合双打男运动员的竞赛优势，Q[i][j]表示女运动员 i 和男运动员 j 配对组成混合双打女运动员的竞赛优势。由于技术配合与心理等各种因素影响，P[i][j]不一定等于 Q[j][i]。男运动员 i 和女运动员 j 配对双打的整体竞赛优势为 P[i][j]*Q[j][i]。请设计算法，计算男女运动员的最佳搭配，使得各组运动员整体竞赛优势总和最大。

要求：

（1）分析该问题适合用何种算法求解。

（2）给出问题解的形式。

（3）给出求解问题的代码框架。

4.1. 把n个任务分给n个人，(i,j)表第i个任务分给第j个人，集合F(i,j)表示不可以将i分给j。

（1）分析该问题适合用何种算法求解。
（2）给出问题解的形式。
（3）给出求解问题的代码框架。

4.2 TSP问题

(1) 上界函数

(2) 下界函数

(3) 结点应包含的信息及理由。(优先队列)

3.2 https://blog.csdn.net/tterminator/article/details/50951137
4.1 https://blog.csdn.net/IOIO_/article/details/81017582
4.2 https://blog.csdn.net/zxzxzx0119/article/details/80055980
3.1 https://wenku.baidu.com/view/b7dfae1128ea81c759f5781a.html
3.3

```
static void(int[]group)
{
int temp;
int pos=0;
for(int i=0;i< group.Length-1;i++)
{
pos=i;
for(intj=i+1;j<group.Length;j++)
{
if(group[j]<group[pos])
{
pos=j;
}
}//第i个数与最小的数group[pos]交换
temp=group[i];
group[i]=group[pos];
group[pos]=temp;
}
}
```

3.1

二、（15 分）

设 A 是 n 个实数的数组，考虑下面的递归算法：

XYZ ($A[1..n]$)

1. if $n=1$ then return $A[1]$
2. else $temp \leftarrow$ XYZ ($A[1..n-1]$)
3. if $temp < A[n]$
4. then return $temp$
5. else return $A[n]$

1. 用简短的文字说明算法 XYZ 的输出是什么？
2. 以 A 中元素的比较作为基本运算，列出算法 XYZ 最坏情况下时间复杂度 $W(n)$ 的递推方程，并解出 $W(n)$。
3. 在求解这个问题的算法类中，算法 XYZ 最坏情况下是不是效率最高的算法？为什么？

1. A 中的最小实数。

2. $W(n)=W(n-1)+1$

   $W(1)=0$

   $W(n)=n-1$

3. 是效率最高的算法，因为找最小问题至少需要比较 $n-1$ 次。

1 利用快速排序 让数组有序化

```
quicksort(*a,left,right){
    i=left   j=right    // 两个工作指针

    if(i<j){            // 直至子表只有一元素（左=右） 排序结束
        pivot=a[i]      //将最左边的字作为比较的关键字

        while(i != j){   //直至左右两个指针遇到 这一趟排序结束

            while(a[j]>=pivotkey)     j--  //先从右边向左找  直至找到一个比它小的
数a[i]
            while(a[i]<=pivotkey)     i++   //再从左边向右找  直到找到一个比它大
的a[j]

            exchange a[i] <--> a[j]

        }
        exchange  a[left] <--> a[i]
    }

    quicksort(*a,left,i-1)
    quicksort(*a,i+1,right)
    //递归调用quicksort 将当前区间以枢轴为界 一分为二
```

O(n log₂n)

2 循环遍历 两两相邻的数的差值 不断更新dvalue

```
different(){
    for(i=2;i<=n;i++)  {      //从数组第二个数开始n-1次遍历
        elem=a[i]-a[i-1]      //不断获得新的差值

        dvalue=elem          //不断获得更小的差值 用dvalue储存
    }
}
```

O(n)

∴ O(n log₂n)

初始化dp，将data的最后一层拷贝到dp中。dp[n][j] = data[n][j] (j = 1, 2, …, n) 其中，n为数塔的层数。
在动态规划过程汇总，我们有dp[i][j] = max(dp[i+1][j], dp[i+1][j+1]) + data[i][j]，最后的结果保存在dp[0][0]中。
对于上面的数塔，我们的data数组如下：

9				
12	15			
10	6	8		
2	18	9	5	
19	7	10	4	16

而我们的dp数组如下：

59				
50	49			
38	34	29		
21	28	19	21	
19	7	10	4	16

下面是C++代码的实现（Visual Studio2013通过）：


```
#include <iostream>
#include <algorithm>

using namespace std;
```

- 当$f(n)$为常数时

$$T(n)=\begin{cases}O(n^{\log_b a}) & a\neq 1\\ O(\log n) & a=1\end{cases}$$

- 当$f(n)=cn$时

$$T(n)=\begin{cases}O(n) & a<b\\ O(n\log n) & a=b\\ O(n^{\log_b a}) & a>b\end{cases}$$