江西师范大学计算机科学技术专业

08-09 第 1 学期《数据结构》期末考试试题 A

课程代号：**262208**
注意事项：请将答案全部写到答题纸上，并注明题号！

一、单项选择题（每小题 2 分，共 12 分）

1．借助结点的存储地址与其值之间映射关系建立的存储结构是（ ）
A．顺序结构 B．链式结构 C．索引结构 D．散列结构

2．给定代码 int i=1, n=100; while(i<n) i=i*2;
其渐进时间复杂度是（ ）
A. O(1) B. O(n) C.O(logn) D.O(nlogn)

3．图的深度优先遍历与树的_____遍历方式相似。
A. 前序遍历 B. 后序遍历 C.层次遍历
D. 没有那种方式与其相似

4．前序遍历和中序遍历序列相同的二叉树，其特点是（ ）
A. 所有结点不能有左子树 B. 所有结点不能有右子树
C. 不含有 1 度结点 D. 该二叉树只有根存在

5．下列排序中，属于稳定的排序方式是（ ）
A．快速排序 B.堆排序 C．希尔排序 D．归并排序

6．队列和栈的主要区别是（ ）
A．逻辑结构不同 B．所包含的运算个数不同
C．存储结构不同 D．限定插入和删除的位置不同

二、程序填空（每空 3 分，共 18 分）

1．在顺序循环队列中查找元素 x，若找到，返回其所在位置；否则，返回-1。

```
#define n 100
typedef  struct{
   int a[n];
   int  front,rear;
}Queue;
int  findX(Queue  *p, int  x){
```

```
    int  i;
    i=   (1)
    while(   (2)    )  i++;
    if (p->a[i]==x)  return i;
    return -1;
}
```

2、下类代码实现 shell 排序。假定所有 len 个关键字已经存储在数组从 a[0] 到 a[len-1]的空间中。

```
    void shell(int a[], int len){
      int i,j,d,t;
      for(d=len/2; d>=1; d=d/2)
         for(   (3)    ; i<len; i++){
           t=a[i];   (4)    ;
           while(   (5)    ){
              (6)    ; j=j-d;
           }
           a[j+d]=t;
      }
    }
```

三、综合解答题（每小题 10 分，共 50 分）

1、给定图 1 所示的有向网络 G，要求：

a. 给出对应的邻接表图；

b. 根据邻接表，写出图的深度、广度优先遍历结果。

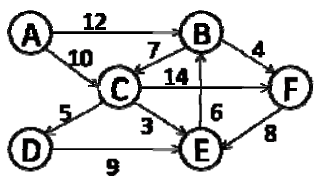


图 1　有向网络 G

2、针对图 1 所示的有向网络 G，基于 Dijkstra 算法将如下表格补充完整。其中 A、…、F 的下标分别对应到 0、…、5。

循环	集合 S	*v*	距离向量 d						路径向量 p					
			0	1	2	3	4	5	0	1	2	3	4	5
初始化	{A}	-												
1														
2														
3														
4														
5														

3、S={12，10，13，23，14，15，17，27，22，33}依次按散列方式存入数组a[10]中，其中散列函数为 H(key)=key%10，冲突处理的方法是线性探测再散列。将下面数组 a 存储各元素的位置示意图填写完整。

	0	1	2	3	4	5	6	7	8	9
数组 a										

4．假设通信电文使用的字符集为{a,b,c,d,e,f,g}，字符的哈夫曼编码依次为：0110，10，110，111，00，0111 和 010。
(1)请根据哈夫曼编码画出此哈夫曼树，并在叶子结点中标注相应字符；
(2)若这些字符在电文中出现的频度分别为：3，35，13，15，20，5 和 9，求该哈夫曼树的带权路径长度。

5、判断（10，30，16，23，19，15，47，27）是否为堆，若不是，将其调整为小根堆（即根最小的堆）。

四、算法设计题（每小题 10 分，共 20 分）

1、给定字符串 s，其长度为 n。s 中只包含英文字母（大/小写）、空格。完成函数 count，统计各字符的个数，放入长度为 27 的数组 t 中，其中 t[0]记录字符'a'和'A'的个数；t[1]记录字符'b'和'B'的个数；……；t[25]记录字符'z'和'Z'的个数；t[26]记录空格的个数。假设数组 t 的所有元素初值已设置为 0（10 分）。

```
count(char *s ,char  t[], int n){
   /*补充完整 */
}
```

2、在二叉排序树 t 中查找元素 x。如 t 为 NULL，生成结点存储元素 x，并将其作为二叉排序树的根；否则，若找到，则无需插入；若未找到，则生成结点存储元素 x，并将该节点插入到二叉排序树的合适位置。（10 分）

```
typedef struct tt{
   int  d;
   struct tt *L,*R;
}btree;
btree* find_insert(btree *t, int  x){
   /*补充完整 */  }
```

江西师范大学计算机科学技术专业

08-09 第 1 学期《数据结构》期末考试试题 A

参考答案

课程代号：**262208**
注意事项：请将答案全部写到答题纸上，并注明题号！

一、单项选择题（每小题 2 分，共 12 分）

1．D　2．C　3．A　4．A　5．D　6．D

二、程序填空（每空 3 分，共 18 分）

1. 在顺序循环队列中查找元素 x，若找到，返回其所在位置；否则，返回-1。

（1）(p->front +1)% n;
（2）i!=rear && p->a[i]!=x

2、下类代码实现 shell 排序。

（3）i=d
（4）j=i-d;
（5）j>=0 && a[j]>t;
（6）a[j+d]=j

三、综合解答题（每小题 10 分，共 50 分）

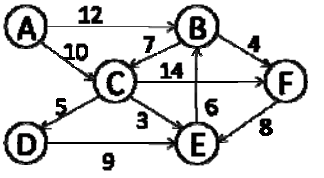
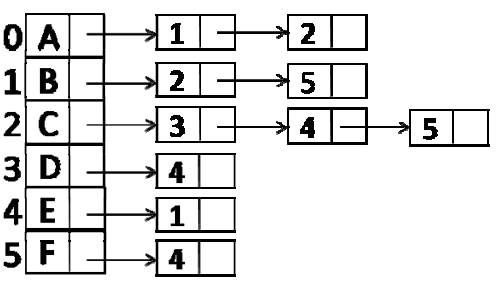


图 1　有向网络 G

1、给定图 1 所示的有向网络 G，要求：

a. 给出对应的邻接表图；
b. 根据邻接表，写出从 A 点开始的深度和广度优先遍历结果。

解答：参考邻接表如下：



深度优先遍历序列为：
A->B->C->D->E->F
广度优先遍历序列为：
A->B->C->F->D->E
注意：邻接表结点的位置可能不同，遍历序列各异。请阅卷老师自行掌控

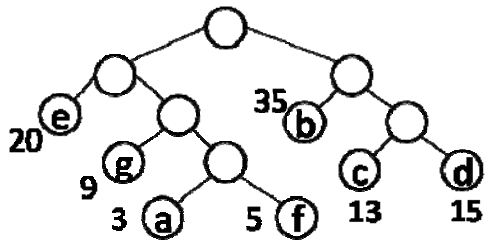
2、针对图 1 所示的有向网络 G，基于 Dijkstra 算法将如下表格补充完整。其中 A、…、F 的下标分别对应到 0、…、5。

循环	集合 *S*	*v*	距离向量 d						路径向量 p					
			0	1	2	3	4	5	0	1	2	3	4	5
初始化	{A}	-	0	12	10	∞	∞	∞	-1	0	0	-1	-1	-1
1	{A,C}	2	0	12	10	15	13	24	-1	0	0	2	2	2
2	{A,C,B}	1	0	12	10	15	13	16	-1	0	0	2	2	1
3	{A,C,B,E}	4	0	12	10	15	13	16	-1	0	0	2	2	1
4	{A,C,B,E,D}	3	0	12	10	15	13	16	-1	0	0	2	2	1
5	{A,C,B,E,D,F}	F	0	12	10	15	13	16	-1	0	0	2	2	1

3、S={12，10，13，23，14，15，17，27，22，33}依次按散列方式存入数组 a[10]中，其中散列函数为 H(key)=key%10，冲突处理的方法是线性探测再散列。将下面数组 a 存储各元素的位置示意图填写完整。

	0	1	2	3	4	5	6	7	8	9
数组 a	10	33	12	13	23	14	15	17	27	22

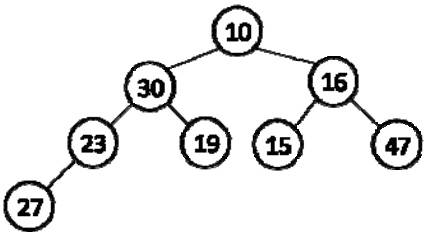
4．假设通信电文使用的字符集为{a,b,c,d,e,f,g}，字符的哈夫曼编码依次为：0110，10，110，111，00，0111 和 010。
(1)请根据哈夫曼编码画出此哈夫曼树，并在叶子结点中标注相应字符；
(2)若这些字符在电文中出现的频度分别为：3，35，13，15，20，5 和 9，求该哈夫曼树的带权路径长度。



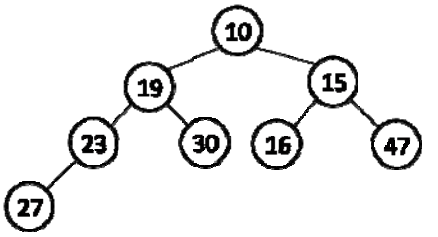
（1）对应的哈夫曼树（6 分）

（2）该哈夫曼树的带权路径长度是：253（4 分）

5、判断（10，30，16，23，19，15，47，27）是否为堆，若不是，先画出初始堆形，然后将其调整为小根堆（即根最小的堆）。



（1）初始堆形　2 分



（2）小根堆　8 分

四、算法设计题（每小题 10 分，共 20 分）

1、给定字符串 s，其长度为 n。s 中只包含英文字母（大/小写）、空格。编写函数，统计各字符的个数，放入长度为 27 的数组 t 中，其中 t[0]记录字符'a'和'A'的个数；t[1]记录字符'b'和'B'的个数；……；t[25]记录字符'z'和'Z'的个数；t[26]记录空格的个数。假设数组 t 的所有元素初值均为 0（10 分）。

```
count (char *s ;char  t[]; int n){
  int i, k;
  for(i=0; i<n; i++){
    if (s[i]>='a'&&s[i]<='z'){k=s[i]-'a'; t[k]++; }
    if (s[i]>='A'&&s[i]<='Z'){k=s[i]-'A'; t[k]++; }
    if (s[i]==' ') t[26]++;
  }
}
```

2、在二叉排序树 t 中查找元素 x。如 t 为 NULL，表示空树，生成结点存储元素 x，并将其作为二叉排序树的根；否则，若找到，则无需插入；若未找到，则生成结点存储元素 x，并将该节点插入到二叉排序树的合适位置。（10 分）

```
typedef struct tt{
   int  d;
   struct tt *L,*R;
}btree;
btree* find_insert(btree *t, int  x){
   btree  *p,*temp;
   p=(btree *)malloc(sizeof(btree));
   p->d=x; p->L=p->R=NULL;
   if ( t==NULL ) {t=p; return t;}
```

```
   if ( t->d==x ) return  t;
   if (x<t->d && t->L==NULL)  {t->L=p;  return  t;}
   if  ( x<t->d && t->L!=NULL){
        temp=find_insert(t->L,x);  return t;}
   if (x>=t->d&& t->R==NULL ){t->R=p; return  t;}
   if (x>=t->d&& t->R!=NULL ){
        temp= find_insert(t->R,x);  return t;}
}
```

注意：本题也可利用递归求解，其阅卷老师自行掌控。

江西师范大学计算机科学技术专业

08-09 第 1 学期《数据结构》期末考试试题 B

课程代号：**262208**
注意事项：请将答案全部写到答题纸上，并注明题号！

一、单项选择题（每小题 2 分，共 12 分）

1．借助结点的存储地址与其值之间映射关系建立的存储结构是（ ）
A．顺序结构 B．链式结构 C．索引结构 D．散列结构

2．给定代码 int i=1, n=100; while(i<n) i=i*2;
其渐进时间复杂度是：
A. O(1) B. O(n) C.O(logn) D.O(nlogn)

3．图的广度优先遍历与树的_____遍历方式相似。
A. 前序遍历 B. 后序遍历 C.层次遍历
D. 没有那种方式与其相似

4．中序遍历和后序遍历序列相同的二叉树，其特点是：
A. 所有结点不能有左子树 B. 所有结点不能有右子树
C. 不含有 1 度结点 D. 该二叉树只有根存在

5．下列排序中，属于不稳定的排序方式是：
A．基数排序 B.堆排序 C．冒泡排序 D．归并排序

6．有 n 个元素的循环队列，f 和 r 分别表示队首和队尾指针，则下列（ ）条件为真时，表示队列中有只有 1 个元素。
A．f+1==r B．(f+1)%n==r C．r+1==f D. (r+1)%n==f

二、程序填空（每空 3 分，共 18 分）

1、下列代码实现稀疏矩阵的转置。

```
typedef  struct {
   int i, j,v; /*i、j、v 分别记录矩阵的行值、列值和元素值*/
}matrix;
/*假设 ma[0]记录矩阵的行数、列数和元素个数 */
void  tran(matrix  ma[], matrix  mb[]){
  int col,p,k;
```

```
    if(ma[0].v==0)  return;
    mb[0].i=ma[0].j;  mb[0].j=ma[0].i;
    mb[0].v=ma[0].v;  k=1;
    for(col=1; （1）; col++)
      for(p=1; p<=ma[0].v; p++)
        if( （2） ) {
            mb[k].i=ma[p].j;
            mb[k].j=ma[p].i;
            mb[k].v=ma[p].v;
            k++;}
  }
```

2、在二叉排序树 t 中查找元素 x。如 t 为 NULL，表示空树，生成结点存储元素 x，并将其作为二叉排序树的根；否则，若找到，则无需插入；否则，生成结点存储元素 x，并将该节点插入到二叉排序树的合适位置。

```
#include  <stdio.h>
typedef struct tt{
  int  d;
  struct tt *L,*R;
}btree;
btree* find_insert(btree *t, int  x){
  btree  *p,*temp;
  p=(btree *)malloc(sizeof(btree));
  p->d=x; p->L=p->R=NULL;
  if ( （3） ) {t=p; return t;}
  if ( （4） ) return  t;
  if (x<t->d && t->L==NULL)  {t->L=p;  return  t;}
  if  ( x<t->d && t->L!=NULL){
      temp=  （5）  ;  return t;}
  if (x>=t->d&& t->R==NULL ){t->R=p; return  t;}
  if (x>=t->d&& t->R!=NULL ){
      temp=  （6）  ;  return t;}
}
```

三、综合解答题（每小题 10 分，共 50 分）

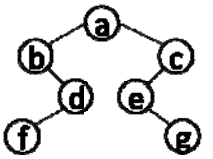


图 1　二叉树 t

1、给定如图 1 所示的二叉树 t，

a.　建立 t 的后序穿线树 t'；

b.　将 t 转化为对应的树或森林。

2、S={12，10，13，23，14，15，17，27，22，33}依次按散列方式存入数组 a[10]中，其中散列函数为 H(key)=key%10，冲突处理的方法是线性探测再散列。将下面数组 a 存储各元素的位置示意图填写完整。

	0	1	2	3	4	5	6	7	8	9
数组 a										

3、给定图 2 所示的有向网络 g，要求基于 Krustral 算法构造 g 的最小生成树，画出构造过程。

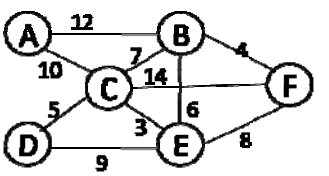


图 2　无向网络 G

4、给定关键字{12，5 ，234，67，58，42，798，567，125，9}，给出静态链式基数排序的过程。

5．从空树起，依次插入关键字 12，10，13，23，14，15，47，27，52，33，构造一棵二叉排序树。

（1）画出该二叉排序树

（2）画出删去该树中元素值为 23 的结点之后的二叉排序树。

四、算法设计题（每小题 10 分，共 20 分）

1、编写函数 findX()，实现在根为 t 的二叉树查找元素 x，若找到，则返回 x 的双亲，否则，返回 NULL。假设 t->d 不等于 x。（10 分）

```
typedef  struct  tt{
   char  d;
   struct  tt *L, *R;
}btree;

btree  *findX(btree *t, int  x){
      /*补充完整 */
}
```

2、给定循环单链表 h，编写函数 delNodes()，实现删除链表中所有值大于 x 而不大于 y 的结点。假设该循环链表带有头结点 h。（10 分）

```
typedef  struct  rr{
   int  d;
  struct  rr *next;
}Ring;
void  delNodes(Ring *h, int  x, int y) {
      /*补充完整 */
}
```

江西师范大学计算机科学技术专业

08-09 第 1 学期《数据结构》期末考试试题 B

参考答案

课程代号：**262208**
注意事项：请将答案全部写到答题纸上，并注明题号！

一、单项选择题（每小题 2 分，共 12 分）

1．D　2．C　3．C　4．B　5．B　6．B

二、程序填空（每空 3 分，共 18 分）

1、下列代码实现稀疏矩阵的转置。

（1）`col<=ma[0].j;`
（2）`ma[p].j==col`

2、在二叉排序树 t 中查找元素 x。如 t 为 NULL，表示空树，生成结点存储元素 x，并将其作为二叉排序树的根；否则，若找到，则无需插入；否则，生成结点存储元素 x，并将该节点插入到二叉排序树的合适位置。

（3）`t==NULL`
（4）`t->d==x`
（5）`find_insert(t->L, x)`
（6）`find_insert(t->R, x)`

三、综合解答题（每小题 10 分，共 50 分）

1、给定如图 1 所示的二叉树 t，

a. 建立 t 的后序穿线树 t'；
b. 将 t 转化为对应的树或森林。

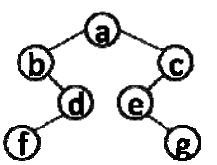
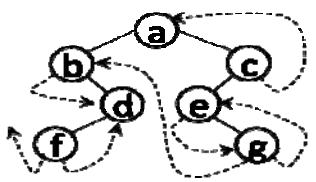
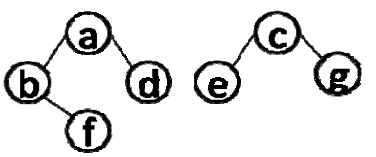


图 1　二叉树 t



a. t 的后序穿线树 t'（5 分）



b. t 对应的森林（5 分）

2、S={12，10，13，23，14，15，17，27，22，33}依次按散列方式存入数组a[10]中，其中散列函数为 H(key)=key%10，冲突处理的方法是线性探测再散列。将下面数组 a 存储各元素的位置示意图填写完整。

	0	1	2	3	4	5	6	7	8	9
数组 a	10	33	12	13	23	14	15	17	27	22

3、给定图 2 所示的有向网络 g，要求基于 Krustral 算法构造 g 的最小生成树，画出构造过程。

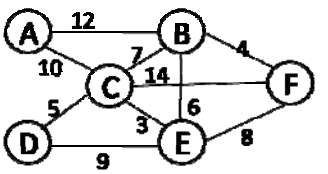
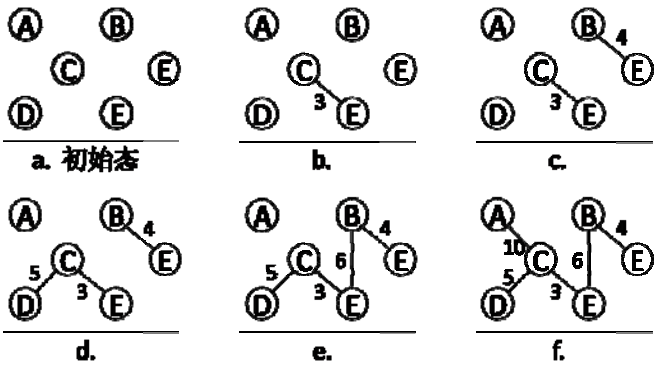


图 2　无向网络 G



b-f 每幅图 2 分

4、给定关键字{12，5 ，234，67，58，42，798，567，125，9}，给出静态链式基数排序的过程。（10 分）

（0）初始状态：12➔5 ➔234➔67➔58➔42➔798➔567➔125➔9

（1）第 1 次分配后的状态：12➔42 ➔234➔5➔125➔67➔567➔58➔798➔9

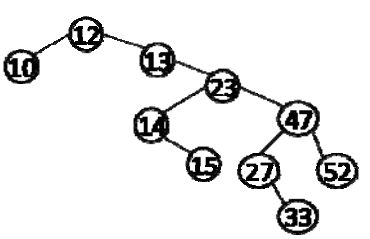
（2）第 2 次分配后的状态：5➔9 ➔12➔125➔234➔42➔58➔67➔567➔798

（3）第 3 次分配后的状态：5➔9 ➔12➔42➔58➔67➔125➔234➔567➔798

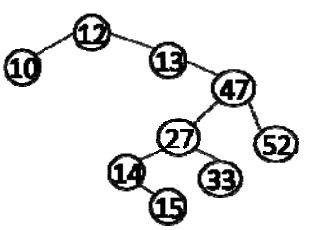
5．从空树起，依次插入关键字 12，10，13，23，14，15，47，27，52，33，构造一棵二叉排序树。（10 分）

（1）画出该二叉排序树

（2）画出删去该树中元素值为 23 的结点之后的二叉排序树。



二叉排序树（5 分）



删除 23 后的二叉排序树（5 分）

四、算法设计题（每小题 10 分，共 20 分）

1、在根为 t 的二叉树查找元素 x，若找到，则返回 x 的双亲，否则，返回 NULL。假设 t->d 不等于 x。（10 分）

```
typedef  struct  tt{
   char  d;
   struct  tt *L, *R;
}btree;
btree  *findX(btree *t, int  x){
   if (t==NULL)  return NULL;
   if(t->L!=NULL && t->L->d==x || t->R!=NULL && t->R->d==x)
     return  t
   if (findX(t->L, x)==NULL)
       return findX(t->R, x);
   return findX(t->L, x);
}
```

2、给定循环单链表 h，删除所有值大于 x 而不大于 y 的结点。假设该循环链表带有头结点 h。（10 分）

```
typedef  struct  rr{
   int  d;
  struct  rr *next;
}Ring;
void  delNodes(Ring *h, int  x, int y){
   Ring  *pre,*p;
   pre=h; p=pre->next;
   while (p!=h)
       if (p->d>x && p->d<=y) {
         p=p->next;  pre->next=p; }
       else{
         pre=p; p=p->next; }
}
```