2010~2011安徽大学《数据结构》期末试卷

一、单选题

从供选择的答案中选出正确的答案，将其编号填入括号中。

1、在数据结构的讨论中把数据结构从逻辑上分为（　　）。

A:　内部结构与外部结构　　B:　静态结构与动态结构

C:　线性结构与非线性结构　　D　紧凑结构与非紧凑结构

2、采用线性链表表示一个向量时，要求占用的存储空间地址（　　）。

A:　必须是连续的　　B　部分地址必须是连续的

C:　一定是不连续的　　C:　可连续可不连续

3、采用顺序搜索方法查找长度为n的顺序表时，搜索成功的平均搜索长度为（　　）。

A:　n　　B:　$n/2$　　C:　$(n-1)/2$　D:　$(n+1)/2$

4、在一个单链表中，若q结点是p结点的前驱结点，若在q与p之间插入结点s，则执行（　　）。

A:　$s\rightarrow link = p\rightarrow link$; $p\rightarrow link = s$;　　**B:**　$p\rightarrow link = s$; $s\rightarrow link = q$;

C:　$p\rightarrow link = s\rightarrow link$; $s\rightarrow link = p$;　　D:　$q\rightarrow link = s$; $s\rightarrow link = p$;

5、如果想在4092个数据中只需要选择其中最小的5个，采用（　　）方法最好。

A:　起泡排序　　B:　堆排序　　C:　锦标赛排序　D:　快速排序

6、设有两个串t和p，求p在t中首次出现的位置的运算叫做（　　）。

A:　求子串　　B:　模式匹配　　C:　串替换　　D:　串连接

7、在数组A中，每一个数组元素$A[i, j]$ 占用3个存储字，行下标i从1到8，列下标j从1到10。所有数组元素相继存放于一个连续的存储空间中，则存放该数组至少需要的存储字数是（　　）。

A:　80　　B:　100　C:　240　D:　270

8、将一个递归算法改为对应的非递归算法时，通常需要使用（　　）。

A:　栈　　B:　队列　　C:　循环队列　　D:　优先队列

9、一个队列的进队列顺序是1, 2, 3, 4，则出队列顺序为（　　）。

A:　4, 3, 2, 1　　B:　2, 4, 3, 1　C:　1, 2, 3, 4　　D:　3, 2, 1, 4

10、在循环队列中用数组*A*[0..*m*−1]　存放队列元素，其队头和队尾指针分别为*front*和*rear*，则当前队列中的元素个数是（　　）。

A: (*front* − *rear* + 1) % *m*　　B: (*rear* − *front* + 1) % *m*

C: (*front* − *rear* + *m*) % *m*　　D: (*rear* − *front* + *m*) % *m*

二、阅读理解题

给出下列递归过程的执行结果。

(1)
```
void unknown ( int w ) {
  if ( w ) {
    unknown ( w-1 );
    for ( int i = 1;  i <= w;  i++ ) cout << w;
  cout << endl;
  }
}
```

调用语句为 *unknown* (4)。

(2)
```
void unknown ( int m ) {
  cout <<  n % 10 ;
  if ( int ( n / 10 ) ) unknown ( int ( n / 10 ) );
}
```

调用语句为*unknown* (582)。

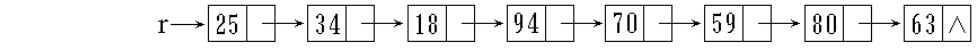
(3)
```
int unknown ( int m ) {
  int value;
  if ( ! m )  value = 3;
  else value = unknown ( m−1 ) + 5;
  return  value;
 }
```

执行语句为 **cout** << *unknown* (3)。

三、填空题

设单链表结构为
```
struct ListNode  {
        int data ;
    ListNode * link ;
};
```

下面的程序是以单链表为存储结构，实现二路归并排序的算法，要求链表不另外占用存储空间，排序过程中不移动结点中的元素，只改各链结点中的指针，排序后r仍指示结果链表的第一个结点。在初始状态下，所有待排序记录链接在一个以r为头指针的单链表中。例如，



在算法实现时，利用了一个队列做为辅助存储，存储各有序链表构成的归并段的链头指

针。初始时，各初始归并段为只有一个结点的有序链表。队列的数据类型为*Queue*，其可直接使用的相关操作有

- 置空队列操作：*makeEmpty* ( );
- 将指针*x*加入到队列的队尾操作：*EnQueue* (*ListNode * x*);
- 退出队头元素, 其值由函数返回的操作：*ListNode *DlQueue* ( );
- 判队列空否的函数, 空则返回1, 不空则返回0：**int** *IsEmpty*( )。

解决方法提示：

➢ 程序首先对待排序的单链表进行一次扫描, 将它划分为若干有序的子链表, 其表头 指针存放在一个指针队列中。

➢ 当队列不空时, 从队列中退出两个有序子链表, 对它们进行二路归并, 结果链表的表头指针存放到队列中。

➢ 如果队列中退出一个有序子链表后变成空队列, 则算法结束。这个有序子链表即为所求。

在算法实现时有 6 处语句缺失，请阅读程序后补上。

(1) 两路归并算法

```
void merge ( ListNode * ha, * hb;  ListNode *& hc ) {
   ListNode *pa, *pb, *pc ;
   if ( ha→data <= hb→data )
         { hc = ha;    pa = ha→link;    pb = hb;  }
   else { hc = hb;    pb = hb→link;    pa = ha ; }
   pc = hc;
   while ( pa && pb )
         if  (  _____(1)_____ ) {
        pc→link = pa;    pc = pa;    _____(2)____ ;
     }
     else {
         pc→link = pb;    pc = pb;    _____(3)____;
     }
    if ( pa )   pc→link = pa;
        else  pc→link = pb;
    };
```

(2) 归并排序主程序

```
void mergesort ( ListNode * r ) {
   ListNode * s, t;   Queue  Q ;
   if  ( ! r )  return;
   s = r;    _____(4)_____ ;
   while ( s ) {
      t = s→link;
```

```
    while ( t != 0 && s→data <= t→data ) { s = t;  t = t→link; }
    if ( t ) {
      s→link = 0; s = t;  _____(5)_____ ;
       }
  }
  while ( !Q.IsEmpty( ) ) {
    r = Q.DlQueue( );
    if ( Q.IsEmpty( ) ) break;
    s = Q.DlQueue( );
    merge( r, s, t );  _____(6)_____ ;
  }
  }
```

四、简答题

(1) 在一个有n个元素的顺序表的第i个元素（$1 \le i \le n$）之前插入一个新元素时，需要向后移动多少个元素？

(2) 当一个栈的进栈序列为1234567时，可能的出栈序列有多少种？6457321是否是合理的出栈序列？

(3) 简单（直接）选择排序是一种稳定的排序方法吗？试举例说明？

(4) 设有序顺序表为 **{** 10, 20, 30, 40, 50, 60, 70 **}**，采用折半搜索时，搜索成功的平均搜索长度是多少？

五、综合算法题

试设计一个实现下述要求的查找运算函数*Locate*。设有一个带表头结点的双向链表*L*，每个结点有4个数据成员：指向前驱结点的指针*llink*、指向后继结点的指针*rlink*，存放字符数据的成员*data*和访问频度*freq*。所有结点的*freq*　　初始时都为0。每当在链表上进行一次*Locate*(*L*, *x*) 操作时，令元素值为*x*的结点的访问频度*freq*加1，并将该结点前移，链接到与它的访问频度相等的结点后面，使得链表中所有结点保持按访问频度递减的顺序排列，以使频繁访问的结点总是靠近表头。