

对外经济贸易大学

2007 年硕士学位研究生入学考试初试试题

考试科目: 841 应用软件基础

注意: 请将所有答案写在答题纸上

第一部分: C 语言程序设计 (共 60 分)

一、选择题 (下列各题 A, B, C, D 四个选项中, 只有一个选项是正确的, 请选择正确的选项) (每小题 1 分, 共 15 分)

1. 有定义: `char a[]="xyz", b[]={'x','y','z'};`, 以下叙述中正确的是【1】

- A) 数组 a 和 b 的长度相同 B) a 数组长度小于 b 数组长度
C) a 数组长度大于 b 数组长度 D) 上述说法都不对

2. 下面程序的运行结果是【2】

```
void f(int *x, int *y)
{ int t;  t=*x; *x=*y; *y=t; }

main()
{ int a[8]={1, 2, 3, 4, 5, 6, 7, 8}, i, *p, *q;
  p=a; q=&a[7];
  while(p<q)
  { f(p, q); p++; q--; }
  for(i=0; i<8; i++) printf("%d, ", a[i]);
}
```

- A) 8, 2, 3, 4, 5, 6, 7, 1, B) 5, 6, 7, 8, 1, 2, 3, 4,
C) 1, 2, 3, 4, 5, 6, 7, 8, D) 8, 7, 6, 5, 4, 3, 2, 1,

3. 下面程序的运行结果是【3】

```
main()
{ char s[]="159", *p;
  p=s;  printf("%c", *p++);
  printf("%c", *p++);
}
```



```

        if(substr[【4】]=='\0') {num++;break;}
    return(num);
}

```

【3】 A) str[i]==substr[i] B) str[i]!='\0'

C) str[i]>substr[i] D) str[i]='\0'

【4】 A) k B) k++ C) k+1 D) ++k

五、编程题（共 15 分）

1. 编写程序实现如下功能：从键盘上输入若干个学生的成绩（假设学生人数不超过 100 人），统计计算出平均成绩，并输出低于平均分的学生成绩，用输入负数结束输入。（8 分）
2. 编写一个函数 creat，用来建立一个带头结点的单向链表，新产生的结点总是插在链表的末尾，单向链表的头指针作为函数值返回。（7 分）

类型和函数首部定义如下：

```

struct list
{
    char data;
    struct list * next;
};

```

struct list * creat(char ch) /*ch 的值作为创建链表时输入结束的标志*/

第二部分：数据库系统（共 60 分）

六、选择题（下列各题 A, B, C, D 四个选项中，只有一个选项是正确的，请选择正确的选项）（每小题 1 分，共 15 分）

1. 数据库管理系统的工作不包括【1】

- A) 定义数据库
- B) 为定义的数据库提供操作系统
- C) 对已定义的数据库进行安全性管理
- D) 数据通信

2. 在数据库中，产生数据不一致的根本原因是【2】

- A) 未对数据进行完整性控制
- B) 数据存储量太大
- C) 没有设置触发器
- D) 数据冗余

3. SQL 语言引入了视图的概念, 下列说法正确的是【3】
- A) 视图由若干表组成, 独立存储在数据库中
 - B) 视图的存在提高了并发程度
 - C) 视图与基本表的最大区别在于它是逻辑定义的虚表
 - D) 视图简化用户观点, 但不提高查询效率
4. NULL 是指【4】
- A) 无任何值
 - B) 0
 - C) 空格
 - D) 空字符串
5. 以下关于外码和相应的主码之间的关系, 正确的是【5】
- A) 外码并不一定要与相应的主码同名
 - B) 外码一定要与相应的主码同名
 - C) 外码一定要与相应的主码同名而且唯一
 - D) 外码一定要与相应的主码同名, 但并不一定唯一
6. 在 SQL 中, 对嵌套查询的处理原则是【6】
- A) 从外层向内层处理
 - B) 从内层向外层处理
 - C) 内、外层同时处理
 - D) 内、外层交替处理
7. 在关系数据库中, 模式对应的是【7】
- A) 索引
 - B) 基本表
 - C) 物理文件
 - D) 存储文件的逻辑结构
8. 关系模式 $R \langle U, F \rangle \in 1NF$, 如果对于 R 的每个非平凡多值依赖 $X \twoheadrightarrow Y (Y \not\subseteq X)$, X 都含有候选码, 则【8】
- A) $R \in 4NF$
 - B) $R \in 3NF$
 - C) $R \in 2NF$
 - D) $R \in 1NF$
9. 系统故障的恢复策略是【9】
- A) 只做 UNDO
 - B) 只做 REDO
 - C) UNDO+REDO
 - D) 先恢复备份, 再做 REDO
10. 审计的作用是【10】
- A) 通过口令校验, 控制用户的访问
 - B) 对数据进行加密, 防止敏感数据泄露
 - C) 记录用户的操作, 事后算帐
 - D) 使用户在进行统计查询时, 不能访问到超过访问权限的数据

11. 如果事务 T 对数据 D 已加 X 锁, 则其他事务对数据 D 【11】
- A) 可以加 S 锁, 不能加 X 锁
 - B) 可以加 S 锁, 也可以加 X 锁
 - C) 不能加 S 锁, 可以加 X 锁
 - D) 不能加任何锁
12. 下列结论哪个是错误的 【12】
- A) $S1 \cup S2 \equiv S2 \cup S1$
 - B) $\pi F(S1 \cap S2) \equiv \pi F(S1) \cap \pi F(S2)$
 - C) $(S1 \cap S2) \cap S3 \equiv S1 \cap (S2 \cap S3)$
 - D) $\pi F(S1 \cup S2) \equiv \pi F(S1) \cup \pi F(S2)$
13. 一个事务中所有对数据库的操作是一个不可分割的操作序列, 这些操作要么全部执行, 要么什么也不做, 该性质是事务的 【13】
- A) 原子性
 - B) 一致性
 - C) 隔离性
 - D) 持久性
14. 逻辑设计的任务是 【14】
- A) 进行数据库的具体定义, 并建立必要的索引文件
 - B) 利用自顶向下的方式进行数据库的逻辑模式设计
 - C) 将概念设计得到的 E-R 图转换为 DBMS 支持的数据模型
 - D) 逻辑模式设计要完成数据的描述, 数据存储格式的设定
15. 规范化过程主要为克服数据库逻辑结构中的插入异常、删除异常以及 【15】 的缺陷。
- A) 冗余度大
 - B) 数据的不一致性
 - C) 结构不合理
 - D) 数据丢失

七、填空题 (每小题 1 分, 共 10 分)

1. 所谓数据库是指长期存储在计算机内的、有组织的、可 【1】 的数据集合。
2. SQL 有两种使用方式: 交互式 and 【2】。
3. 等式 $R \bowtie S = R \times S$ 成立的条件是 【3】。
4. 数据模型的三要素是: 数据结构、【4】和完整性约束。
5. 关系数据库用 【5】 实现数据之间的联系。
6. 在一个关系 R 中, 若每个数据项都是不可再分割的, 那么 R 属于 【6】 NF。

7. 表示概念模型的最常用的方法是【7】。
8. 目前最常用的数据模型除了层次模式、网状模型、关系模型外，还有【8】。
9. 封锁机制是实现数据库【9】的主要方法。
10. 数据库系统结构中，改变模式但可保持应用程序不变的特性称为【10】。

八、判断题（认为表述正确，填写“T”，否则填“F”，每小题 1 分，共 6 分）

1. 【1】 3NF 不仅要取消非主属性对主属性的部分依赖和传递依赖，还要取消主属性对码的部分依赖和传递依赖。
2. 【2】 函数依赖由语义决定，与属性值无关。
3. 【3】 SQL 中，与“NOT IN”等价的操作符是“<>ALL”
4. 【4】 关系数据模型的优点主要有：建立在严格的数学概念基础上；关系模型概念单一；关系模型存取路径对用户透明。
5. 【5】 遵循三级封锁协议，肯定遵循两段锁协议。
6. 【6】 关系数据库中，关系模式是型，关系的逻辑表达是值。

九、名词解释（每小题 2 分，共 4 分）

1. 相关子查询
2. 实体完整性规则

十、综合题（共 3 题，共 25 分）

1. （本小题共 5 分）假设学生选课数据库关系模式如下：

学生 Student (SNO, SNAME, SAGE, SDEPT)

课程 Course (CNO, CNAME)

选课 SC (SNO, CNO, SCORE)

对应的中文含义为：学生 Student（学号，姓名，年龄，系别）

课程 Course（课程号，课程名）

选课 SC（学号，课程号，课程成绩）

用关系代数进行如下查询：

(1) 计算机系有不及格课程的学生名单； (2 分)

(2) 学生刘国星的“数据库系统”课程成绩。 (3 分)

2. （本小题共 10 分）某公司有多名销售人员负责公司的商品销售业务，每名客户可以一次性订购多种商品，每件商品都由惟一的商品号标识，下表为销售商品

的详细订单。

商品订购单

订单号	1293463	日期	02/20/06	付款方式	现金支付	总金额	2400.00
客户号	11020478	客户姓名	张娜		联系电话	010—64491234	
地址	北京市朝阳区惠新东街 10 号				邮政编码	100029	
商品号	商品列表		规格		单价/元	总计/元	
110001	TCL29 寸纯平彩电		T168		2500.00	2500.00	
110018	欧亚达沙发		H901		700.00	700.00	
120032	爱仕达不锈钢蒸锅		V1528		229.50	229.50	
销售人员号	3240	销售人员姓名	李永强		电话号码	13301023456	

(1) 试为该公司的商品销售业务数据库设计 E-R 图, 满足 3NF。

(2) 将 E-R 图转换为关系模式集, 并标出每个关系模式的主键和外键。

3. (本小题 10 分) 设某花店销售可以在不同地区生长的花, 这些地区一年的最低温度在一定范围内变化, 该数据库中共建有 3 张基本表:

地区表 ZONE (Z_ID, LowerTemp, UpperTemp)

发货表 DELIVERY (D_ID, Category, DelSize)

鲜花表 FLOWERINFO (F_ID, ComName, Czone, Hzone, D_ID, SunNeeds)

属性含义如下:

Z_ID: 地区号, 字符型, 范围 01-99, 地区号越小代表最低温度越低

LowerTemp: 一年中地区最低温度范围低限, 数值型

UpperTemp: 一年中地区最低温度范围高限, 数值型

D_ID: 发货号, 字符型

Category: 发货类型, 包括 pot, plant, hedge, shrub, tree 等, 字符型

DelSize: 发货规格, 数值型

F_ID: 鲜花号, 字符型

ComName: 花的名称, 字符型

Czone: 最冷地区号, 字符型

Hzone: 最热地区号, 字符型

SunNeeds: 日光需求, 字符型

请写出完成下列要求的 SQL 语句:

(1) 将 2 号地区最低温度范围为-50 度至-40 度和 5 号地区最低温度范围为-20 度至-10 度两组数据插入 ZONE 表中。

- (2) 列出可以在 9 号地区生长的所有鲜花的 ID 和名称。
- (3) 查询以花瓶(pot)形式发货的所有鲜花的 ID、名称以及发货规格, 结果按鲜花名称升序排列。
- (4) 查询以 plant 或 tree 形式发货的所有鲜花的名称及其可以生长的最低温度。
- (5) 鲜花公司停止销售鲜花' Lady Fern', 从表中删除。

第三部分: 计算机网络 (共 30 分)

十一、选择题 (下列各题 A, B, C, D 四个选项中, 只有一个选项是正确的, 请选择正确的选项) (每小题 1 分, 共 10 分)

1. 目前, 我国已经开通的下一代互连网的实验网使用的网络互连协议名为【1】。
A) IPV4 B) IPV6 C) SLIP D) PPP
2. Internet 网络是由若干【2】组成的。
A) 局域网 B) 广域网 C) 网络 D) 都不对
3. OSI 协议体系结构由【3】层协议组成。
A) 5 B) 7 C) 8 D) 4
4. 在局域网中, 计算机之间的数据采用【4】传输方式。
A) 一对多 B) 并行和串行 C) 串行 D) 并行
5. 在网络上, 采用【5】交换技术更适合传输较小的数据块。
A) 虚电路 B) 数据报 C) 电路 D) 都对
6. 在 CSMA/CD 原理中, CSMA 的作用是解决【6】网络中数据发送的碰撞问题。
A) 广播式 B) 互连的 C) 共享传输介质的 D) 都对
7. 链路是指【7】节点之间的物理线路。
A) 相邻的 B) 同一网络的 C) 源-目的 D) 不同网络中
8. 在本地网络中, 靠【8】地址识别计算机。
A) IP B) TCP C) MAC D) PORT
9. 物理层的四个特性中, 【9】特性决定了每段网线的长度和速度。
A) 机械 B) 规程 C) 电气 D) 功能
10. 利用同步技术保证【10】。
A) 接收完整的数据 B) 避免重复发送
C) 避免编码出错 D) 传输速度快

十二、填空题（每空 1 分，共 10 分）

1. 面向连接的服务，必须在传输之前建立【1】，帧按建立的固定路线发送。
2. 如果一个路由器连接 N 个不同网络，那么这个路由器具有【2】个 IP 地址。
3. 时分多路复用技术将传输的时间分成若干时间片，分配给不同的【3】对。
4. DNS 服务器的作用是将【4】解析成 IP 地址。
5. ATM 网络中的 ATM 层 CELL 长度为 53【5】。
6. 1000 BASE-LX 是高速以太网的一个标准，BASE 是指【6】。
7. 发送的数据被拒绝的原因之一是发送的数据包所携带的【7】不对，导致找不到接收者。
8. 在网络层，路由器不但要承担路由选择功能，还要具有从端口接收和【8】包的功能。
9. 以太网的交换机的每个端口相当于一个计算机的【9】。
10. FTP 服务器使用【10】号端口接收 FTP 请求。

十三、综合题（共 10 分）

1. 一个提供 B TO C 电子商务的网站 A，对存放商品、客户数据的服务器有较高的性能需求，从网络服务的角度，请列举你认为最重要的 2 项服务器的技术指标，并做简单说明。（2 分）
2. 对于 1 中提到的网站 A，采用 B/S 还是 C/S 结构更好一些，并阐述理由。（3 分）
3. 对于 1 中提到的网站 A，你认为还应该配置哪些服务器，请列举出来。（3 分）
4. 假设上述 A 网站正在运行中，突然发现有大量的请求信息来自同一个外部 IP 地址，你认为可能出现了什么情况？（2 分）

- A) 15 B) 16 C) 26 D) 59

4. 下面函数的功能是【4】

```
fun(char *a, char *b)
{while((*a!='\0')&&(*b!='\0')&&(*a==*b)) {a++;b++;}
return(*a-*b);
}
```

- A) 计算 a 和 b 所指字符串的长度之差
B) 将 b 所指字符串连接到 a 所指字符串
C) 将 b 所指字符串连接到 a 所指字符串后面
D) 比较 a 和 b 所指字符串的大小
5. 有程序段: int k=0; while(k=1)k++; , while 循环执行的次数是【5】
A) 无限次 B) 有语法错, 不能执行
C) 一次也不执行 D) 执行 1 次
6. 有说明: int a[10]={1, 2, 3, 4, 5, 6, 7, 8, 9, 10}, *p=a;. 则数值为 9 的表达式是【6】
A) *p+9 B) *(p+8) C) *p+=9 D) p+8
7. 下面程序的输出结果是【7】

```
#include <stdio.h>
static int a=50;
f1( int a )
{ printf( " %d ,", a+=10 ); }
f2( )
{printf( " %d ,", a+=3 ); }
main( )
{ int a=10; f1 (a); f2( ); printf( "%d\n", a); }
```

- A) 60, 63, 60 B) 20, 23, 23 C) 20, 13, 10 D) 20, 53, 10
8. 已知职工记录描述如下, 设变量 w 中的“生日”是“1993 年 10 月 25 日”, 下列对“生日”的正确赋值方式是【8】


```
struct worker
{
    int no; char name[20]; char sex;
    struct birth
    {
        int day; int month; int year;
    } a;
};
```

```
struct worker w;
```

- A) day=25; month=10; year=1993;
- B) w.birth.day=25; w.birth.month=10; w.birth.year=1993;
- C) w.day=25; w.month=10; w.year=1993;
- D) w.a.day=25; w.a.month=10; w.a.year=1993;
9. 关于 return 语句, 下列正确的说法是【9】
- A) 可以在同一函数中出现多次
- B) 在主函数中不能出现
- C) 必须在每个函数中出现
- D) 只能在除主函数之外的函数中出现一次
10. 要打开一个已存在的非空文件“file”用于修改, 选择正确的语句【10】
- A) fp=fopen("file", "r"); B) fp=fopen("file", "w");
- C) fp=fopen("file", "r+"); D) fp=fopen("file", "w+");
11. 已知职工记录描述如下, 在 Turbo C 中, 系统为变量 w 分配的字节空间数为【11】

```
struct worker
{
    int no;
    char name[20];
    char sex;
    union
    {
        int day; int month; int year;
    } birth;
} w;
```

- A) 29 B) 25 C) 20 D) 6

12. 现有以下结构体说明和变量定义, 如图 1 所示, 指针 p, q, r 分别指向一个链表中连续的两个结点。

```
struct node
{
    char data;
    struct node *next;
} *p, *q, *r;
```

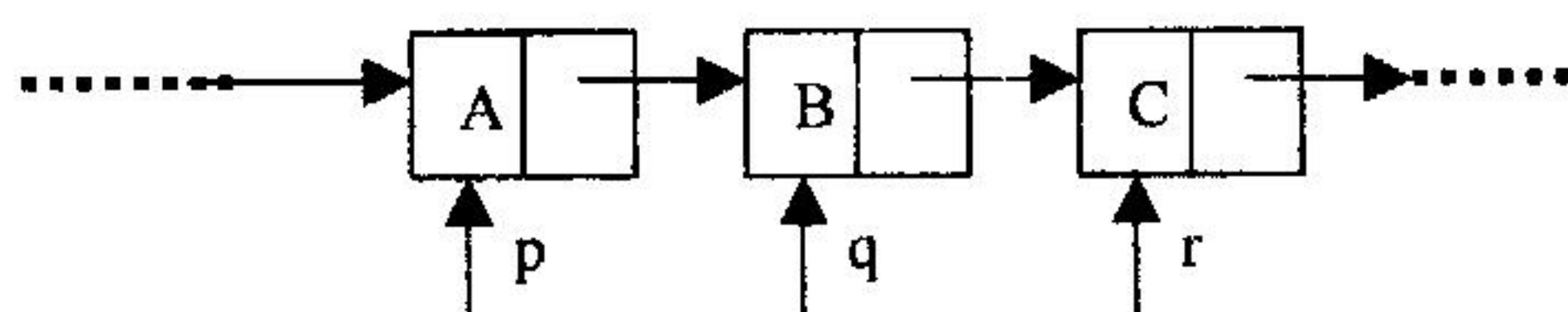


图 1 选择题 12 题图

现要将 q 和 r 所指结点交换前后位置, 同时要保持链表的连续, 以下不能完成此操作的语句是【12】

- A) $q \rightarrow next = r \rightarrow next; p \rightarrow next = r; r \rightarrow next = q;$
- B) $p \rightarrow next = r; q \rightarrow next = r \rightarrow next; r \rightarrow next = q;$
- C) $q \rightarrow next = r \rightarrow next; r \rightarrow next = q; p \rightarrow next = r;$
- D) $r \rightarrow next = q; p \rightarrow next = r; q \rightarrow next = r \rightarrow next;$

13. 以下叙述正确的是【13】

- A) 预处理命令行必须位于 C 源程序的起始位置
- B) 在 C 语言中, 预处理命令行都以“#”开头
- C) 每个 C 程序必须在开头包含预处理命令行: `#include <stdio.h>`
- D) C 语言的预处理不能实现宏定义和条件编译的功能

14. 以下程序段中与语句 $k = a > b ? (b > c ? 1 : 0) : 0;$ 功能等价的是【14】

- | | |
|---|--|
| A) <code>if((a>b)&&(b>c)) k=1;</code> | B) <code>if((a>b) (b>c)) k=1;</code> |
| <code>else k=0;</code> | <code>else k=0;</code> |
| C) <code>if(a<=b) k=0;</code> | D) <code>if(a>b) k=1;</code> |
| <code>else if(b<=c) k=1;</code> | <code>else if(b>c) k=1;</code> |
| | <code>else k=0;</code> |

15. 有以下程序

```
#include <stdlib.h>

struct NODE{ int num;
             struct NODE *next;
             };

main( )
{ struct NODE *p, *q, *r;
  int sum=0;
  p=(struct NODE *)malloc(sizeof(struct NODE));
  q=(struct NODE *)malloc(sizeof(struct NODE));
  r=(struct NODE *)malloc(sizeof(struct NODE));
  p->num=1;q->num=2;r->num=3;
  p->next=q;q->next=r;r->next=NULL;
  sum+=q->next->num;sum+=p->num;
  printf("%d\n", sum);
}
```

执行后的输出结果是【15】

- A) 3 B) 4 C) 5 D) 6

二、写出程序的运行结果（每小题 2 分，共 10 分）

1. 下面程序的输出结果是【1】

```
hcf(int u,int v)
{ int a,b,c,r;
  if(u>v)    { c=u;  u=v;  v=c; }
  a=u; b=v;
  while((r=b%a)!=0)  { b=a; a=r;}
  return(a);
}

main()
{int u,v,h,l;
```



```
u=32; v=48;
h=hcf(u, v);
printf("H. C. F=%d\t", h);
}
```

2. 下面程序的输出结果是【2】

```
main()
{ int k=0; char c=' A' ;
  do{switch(c++)
    { case 'A': k++;break;
      case 'B': k--;
      case 'C': k+=2;break;
      case 'D': k=k%2;continue;
      case 'E': k=k+10;break;
      default: k=k/3;
    }
  }while(c<' C' );
  printf( "k=%d\n", k);
}
```

3. 下面程序的输出结果是【3】

```
void ive(int x[], int n)
{ int t, *p;
  p=x+n-1;
  while (x<p)
    { t=*x; *x++=*p; *p--=t;}
}

main()
{ int i, a[]={1, 2, 3, 4, 5, 6, 7, 8, 9, 0};
  ive(a, 10);
  for(i=0; i<10; i++) printf( "%4d", a[i]);
}
```



```
printf( "\n" );
}
```

4. 下面程序的输出结果是【4】

```
main()
{ int i=10, a[]={10, 20, 30, 15}, *p1;
  char *b=" Learning", **p2=&b;
  p1=&i;
  printf( "%4d", *p1+20);
  for(p1=a;p1<a+4;p1+=2) printf( "%4d", *p1);
  printf( "\n%c\t%s\n", *(b+4), *p2);
}
```

5. 下面程序的输出结果是【5】

```
struct ks
{ int a; int *b; }
main()
{ struct ks s[4], *p; int n=1, i ;
  for (i=0; i<4; i++)
  { s[i].a=n; s[i].b=&s[i].a; n=n+2; }
  p=&s[0]; printf( "%d,%d\n", ++(*p->b), *s[2].b);
}
```

三、程序填空题（每空 2 分，共 12 分）

1. 假设 a 数组中的数据已按照由大到小的顺序存放，下面的程序可把 a 数组中相同的数据删除后只剩一个，然后输出 a 数组中的数据，请填空。

```
#define M 10
main()
{ int a[M], i, j, n;
  for (i=0; i<M; i++) scanf( "%d", a+i);
  n=i=M-1;
  while(i>=0)
```



```

        { if (*(a+i)==*(a+i-1))
            { for(j=【1】;j<=n;j++) *(a+j-1)=*(【2】);
              n--;
            }
          i--;
        }
    }

    for(i=0;i<=n;i++) {printf( "%4d",a[i]);}
}

```

2. 以下函数把 b 字符串连接到 a 字符串的后面，并返回 a 中新字符串的长度。
请填空。

```

Strcen(char a[], char b[] )
{ int m=0, n=0;
  while(*(a+ m)!=【3】) m++;
  while( b[n] ){ *(a+ m)=b[n]; m++;【4】; }
  return(m);
}

```

3. 下面程序的功能是找出数组 x 中的最大值和该值所在的下标，数组元素值从键盘输入。请填空。

```

main()
{ int x[10], *p1, *p2, k;
  for ( k=0; k<10; k++) scanf( "%d", &x[k]);
  for( p1=x, p2=x; p1-x<10; p1++)
    if ( *p1>*p2 ) p2=【5】;
  printf( "max=%d, index=%d", *p2,【6】);
}

```

四、程序选择填空（每空 2 分，共 8 分）

1. 下面的程序是将字符串 s 插入到已排好序（升序）的字符串 a 中，且插入后 a 仍然是有序的。请选择填空。


```

#include <stdio.h>

main()
{
    char a[20]="cehiknqtw";
    char s[ ]="fbla"; int i, j, k;
    for (k=0; s[k]!='\0'; k++)
    {
        j = 0;
        while(s[k]>=a[j] && a[j]!='\0') j++;
        for (【1】) 【2】;
        a[j]=s[k];
    }
    puts(a);
}

```

【1】 A) i=strlen(a)+k; i>=j; i--

B) i=strlen(a); i>=j; i--

C) i=j; i<=strlen(a)+k; i++

D) i=j; i<=strlen(a); i++

【2】 A) a[i]=a[i+1]

B) a[i+1]=a[i]

C) a[i]=a[i-1]

D) a[i-1]=a[i]

2. 下面程序的功能是统计子串 substr 在母串 str 中出现的次数。请选择填空。

```

#include "stdio.h"

main()
{
    char str[80], substr[80]; int n;
    gets(str); gets(substr);
    printf( "%d\n", count(str, substr));
}

count(char *str, char *substr)
{
    int i, j, k, num=0;
    for(i=0; 【3】 ;i++)
        for(j=i, k=0; substr[k]==str[j]; k++, j++)

```