

(2023 年) 广东省中山市全国计算机等级 考试网络技术测试卷(含答案)

学校:_____ 班级:_____ 姓名:_____ 考号:_____

一、单选题(10 题)

(16) TCP/IP 参考模型中, 下列关于应用层的描述不正确的是

1.
 - A) 向用户提供一组常用的应用程序
 - B) 是 ISO/OSI 参考模型的应用层、表示层和会话层
 - C) 位于 TCP/IP 参考模型中的最高层
 - D) 负责相邻计算机之间的通信

2. 下列关于 B/S 模式应用服务器的描述中, 错误的是()

- A. 网络应用建立在 Web 服务的基础上
- B. 访问不同的应用服务器需要不同的客户端程序
- C. 浏览器不能直接访问数据库服务器
- D. 采用 3 层架构

3. 奈奎斯特定理描述了有限带宽、无噪声信道的最大数据传输速率与信道带宽的关系。对于二进制数据, 若最大数据传输速率为 6000bps, 则信道带宽 B=()。

- A. 300Hz B. 6000Hz C. 3000Hz D. 2400Hz

4. 下列关于综合布线的描述中, 正确的是 ()。

- A. 多介质插座是用来连接 UTP 和 STP 的

- B.适配器具有转换不同数据速率的功能
- C.与 STP 相比，UTP 防止对外电磁辐射的能力更强
- D.对于建筑群子系统来说，巷道布线法是最理想的方式

(40) HTML 语言的特点包括

- A)通用性、简易性、可扩展性、平台无关性
 - B)简易性、可靠性、可扩展性、平台无关性
 - C)通用性、简易性、真实性、平台无关性
 - D)通用性、简易性、可扩展性、安全性
- 5.

6.针对不同的传输介质，Ethernet 网卡提供了相应的接口，其中适用于非屏蔽双绞线的网卡应提供()。

- A.AUI 接口 B.BNC 接口 C.RS-232 接口 D.RJ-45 接口

7.第 15 题 关于划分 OSI 参考模型层次的原则是 ()。

- . 网中各结点都有相同的层次
- . 不同结点的同等层具有相同的功能
- . 同一结点相邻层之间通过接口通信
- . 每一层使用下层提供的服务，并向其上层提供服务
- . 不同结点的同等层按照协议实现对等层之间的通信

- A.、、 B.、、 C.、、、 D.都是

8.进行中长期天气预报属于计算机哪个应用领域()

- A.科学计算 B.事务处理 C.过程控制 D.人工智能

9.关于 UNIX, 以下说法错误的是()。

- A.支持多任务和多用户
- B.提供可编程 Shell 语言
- C.系统全部采用汇编语言编写而成, 运行速度快
- D.采用树形文件系统, 安全性和可维护性良好

10.下列说法错误的是()。

- A.奔腾是 32 位的, 这里 32 位指的是处理器能够保存 32 位的数据
- B.计算机的位数不是指十进制数, 而是十六进制数
- C.通常称 8 位是一个字节(Byte), 16 位是一个字(Word)
- D.用于安腾处理器上的芯片已经是 64 位了

二、填空题(10 题)

11.

12. 一台主机的 IP 地址为 10.1.1.100, 屏蔽码为 255.0.0.0。现在用户需要配置该主机的默认路由。如果与该主机直接相连的惟一的路由器具有两个 IP 地址, 一个为 10.2.1.100, 屏蔽码为 255.0.0.0, 另一个为 11.1.1.1, 屏蔽码为 255.0.0.0, 那么该主机的默认路由应该为_____。

13.Telnet 协议使用提供一种标准的键盘定义, 从而屏蔽不同系统对键盘

定义差异。

14.(4) 在星型拓扑结构中，_____节点是全网可靠性的瓶颈

15. 计算机病毒的完整的工作过程包括潜伏、_____、触发和行为控制四个阶段。

16. 通信控制处理机可作为通信子网中的_____，完成分组的接收、校验、存储及转发等功能，实现将源主机报文准确发送到目的主机的作用。

17.

18. Google 搜索引擎主要采用了分布式爬行网页采集技术、超文本匹配技术和_____。

19. 传输层的服务访问点是。

20.(20) IP 电话系统有 4 个基本组件：终端设备、_____、多点控制单元和网守。

三、2. 填空题(8 题)

21. 网卡是连接局域网中计算机和【 】的设备。

22. 网状型拓扑结构又称无规则型结构，结点之间的连接是【 】。
23. FastEthernet 的数据传输速率为 100Mbps，保留着与传统的 10Mbps 速率 Ethernet 的帧格式。
24. 安腾是_____位的芯片。
25. _____涉及修改数据流或创建数据流，它包括假冒、重放、修改消息和拒绝服务等。
26. HP-UX 是_____公司的 UNIX 系统。
27. 常用的电子支付工具包括【 】、电子信用卡和电子支票。
28. 双绞线用于 10Mb/s 局域网时，与集线器的距离最大为【 】m。

四、C 语言程序设计题(2 题)

29. 已知数据文件 in.dat 中存有 300 个四位数，并已调用读函数 readDat 把这些数存入数组 a 中，请编制一函数 isValue，其功能是：求出所有这些四位数是素数的个数 cnt，再把所有满足此条件的四位数依次存入数组 b 中，然后对数组 b 的四位数按从小到大的顺序进行排序。最后 main 函数调用写函数 writeDat 把数组 b 中的数输出到 out.dat 文件中。

例如 :5591 是素数, 则该数满足条件存入数组 b 中, 且个数 $cnt=cnt+1$ 。

9812 是非素数, 则该数不满足条件忽略。

注意 : 部分源程序存在 test . C 文件中。

程序中已定义数组 : a[300], b[300], 已定义变量 : cnt 请勿改动数据文件 in . dat 中的任何数据、主函数 main、读函数 readDat 和写函数 writeDat 的内容。

```

1  #include <stdio.h> //include 语句说明各程序中包含 vc6.0 中的标准输入输出库函数 stdio.h
2  int a[300], b[300], cnt=0; //定义全局数组 a[300], b[300] 和变量 cnt, 并且对变量 cnt 赋初值 0
3  void readDat(); //函数 readDat() 说明语句
4  void writeDat(); //函数 writeDat() 说明语句
5  void jsValue(); //函数 jsValue() 说明语句
6  int isP(int m) //函数 isP(m) 判断 m 是否为素数, 如果是素数, 返回 1, 否则返回 0
7  {
8      int i; //定义变量 i
9      for(i = 2; i < m; i++) //循环变量 i 从 2 依次递增, 直到 i 等于或大于 m 退出循环
10     if(m % i == 0)
11         return 0; //如果 m 能被 i 整除, 返回 0
12     return 1; //否则返回 1
13 }
14 void jsValue()
15 {
16 }
17 }
18 void main()
19 {
20     int i; //定义变量 i
21     readDat(); //调用 readDat() 函数从数据文件 in.dat 中读取 300 个四位数存入数组 a 中
22     jsValue(); //调用函数 jsValue() 实现题目所要求的功能
23     writeDat(); //调用 writeDat() 函数把计算结果写入到数组 b 中的数输出到 out.dat 文件
24     printf("cnt=%d\n", cnt); //在屏幕上显示素数的个数
25     for(i = 0; i < cnt; i++)
26         printf("b[%d]=%d\n", i, b[i]); //在屏幕上显示数组 b 中的所有元素
27 }
28 void readDat()
29 {
30     FILE *fp; //定义文件指针变量 fp
31     int i; //定义整型变量 i
32     fp = fopen("in.dat", "r"); //以只读的方式打开文件 in.dat, 并用 fp 指向这个文件
33     for(i = 0; i < 300; i++)
34         fscanf(fp, "%d,", &a[i]); //从文件 in.dat 中读取 300 个四位数到数组 a 中
35     fclose(fp); //关闭文件 in.dat
36 }
37 void writeDat()
38 {
39     FILE *fp; //定义文件指针变量 fp
40     int i; //定义整型变量 i
41     fp = fopen("out.dat", "w"); //以只写的方式打开文件 out.dat, 并用 fp 指向这个文件
42     fprintf(fp, "%d\n", cnt); //把素数的个数写入到文件 out.dat
43     for(i = 0; i < cnt; i++)
44         fprintf(fp, "%d\n", b[i]); //把数组 b 中的所有元素写入到文件 out.dat
45     fclose(fp); //关闭文件 out.dat
46 }

```

30. 已知在文件 in . dat 中存有 100 个产品销售记录, 每个产品销售记录由产品代码 dm(字符型 4 位), 产品名称 mc(字符型 10 位), 单价 dj(整型), 数量 sl(整型), 金额 je(长整型)五部分组成。其中: 金额=单价*数

量。函数 `ReadDat` 读取这 100 个销售记录并存入结构数组 `sell` 中。请编制函数 `SortDat`，其功能要求：按产品代码从大到小进行排列，若产品代码相同，则按金额从大到小进行排列，最终排列结果仍存入结构数组 `sell` 中。最后 `main` 函数调用函数 `WriteDat` 把结果输出到文件 `out.dat` 中。

提示：若中间变量为 `PRtemp`，则可以直接使用结构赋值语句进行解题；产品代码比较请用函数 `strcmp` 进行解题。

例如：`sell[i]=temp;`

注意：部分源程序存放在 `test.c` 文件中。

请勿改动主函数 `main`、读函数 `ReadDat` 和输出函数 `WriteDat` 的内容。


```

1  #include <stdio.h>
2  #include <string.h>
3  #include <stdlib.h>
4  #define MAX 100
5  typedef struct
6  {
7      char dm[5]; /* 产品代码 */
8      char mc[11]; /* 产品名称 */
9      int dj; /* 单价 */
10     int sl; /* 数量 */
11     long je; /* 金额 */
12 } PRO; //定义结构体 PRO
13 PRO sell[MAX]; //定义结构体一维数组变量 sell[MAX]
14 void ReadDat();
15 void WriteDat();
16 void SortDat()
17 {
18
19 }
20 void main()
21 {
22     memset(sell, 0, sizeof(sell)); //为 sell 分配 sizeof(sell) 大小的内存空间
23     ReadDat();
24     SortDat();
25     WriteDat();
26 }
27 /*读取这100个销售记录并存入结构数组 sell 中*/
28 void ReadDat()
29 {
30     FILE *fp;
31     char str[80], ch[11];
32     int i;
33     fp = fopen("in.dat", "r");
34     for(i = 0; i < 100; i++)
35     {
36         fgets(str, 80, fp); //从文件 in.dat 中读取长度为 79 的字符串存入字符数组 str 中
37         memcpy(sell[i].dm, str, 4); //从字符串 str 的开始位置取长度为 4 的字符串赋给产品代码
38         memcpy(sell[i].mc, str + 4, 10); //从字符串 str+4 的位置取长度为 10 的字符串赋给产品名称
39         memcpy(ch, str + 14, 4); //从字符串 str+14 的位置取长度为 4 的字符串赋给字符数组 ch

```

```

39     ch[4] = 0; //把零赋给字符数组元素 ch[4]
40     sell[i].dj = atoi(ch); //把字符数组 ch 转化成整型数值赋给产品单价
41     memcpy(ch, str + 18, 5); //从字符串 str+18 的位置取长度为 5 的字符串赋给字符数组 ch
42     ch[5] = 0; //把零赋给字符数组元素 ch[5]
43     sell[i].sl = atoi(ch); //把字符数组 ch 转化成整型数值赋给产品数量
44     sell[i].je = (long)sell[i].dj * sell[i].sl; //产品单价乘以产品数量等于产品金额
45 }
46 fclose(fp);
47 }
/*把结果输出到文件 out.dat 中*/
48 void WriteDat()
49 {
50     FILE *fp;
51     int i;
52     fp = fopen("out.dat", "w");
53     /*把经过处理的 100 条记录写入到文件 out.dat 中*/
54     for(i = 0; i < 100; i++)
55     {
56         fprintf(fp, "%s %s %4d %5d %10ld\n", sell[i].dm, sell[i].mc, sell[i].dj, sell[i].sl,
57             sell[i].je);
58     }
59     fclose(fp);
60 }

```

五、1.选择题(3 题)

31. 关于 B-ISDN 说法中不正确的是_____。

- A.在 B-ISDN 中，用户环路和干线一般采用光缆
- B.B-ISDN 采用快速分组交换与 ATM，而不采用电路交换
- C.B-ISDN 中使用虚通路的概念，其比特率不预先确定
- D.B-ISDN 只能向用户提供 2M 以下的业务

32.从用户的角度看，因特网是一个()。

- A.广域网 B.远程网 C.综合业务服务网 D.信息资源网

33. Elgamal 公钥体制是一种基于离散对数的 Elgamal 公钥密码体制，
又称其为_____。

- A.背包公钥体制 B.数据签名标准 C.椭圆曲线密码术 D.概率加密体制

六、1.程序设计题(3 题)

34. 对 10 个候选人进行选举，现有一个 100 条记录的选票数据文件 IN29.DAT，其数据存放的格式是每条记录的长度均为 10 位，第一位表示第一个人的选中情况，第二位表示第二个人的选中情况，依次类推。每一位内容均为字符 0 或 1，1 表示此人被选中，0 表示此人未被选中，若一张选票选中人数小于等于 5 个人时则被认为是无效的选票。给定函数 ReadDat()的功能是把选票数据读入到字符串数组 xx 中。请编制函数 CountRs()来统计每个人的选票数并把得票数依次存入 yy[0]到 yy[9]中，最后调用函数 WriteDat()把结果 yy 输出到文件 OUT29.DAT 中。

注意：部分源程序已给出。

请勿改动主函数 main()、读函数 ReadDat()和写函数 WriteDat()的内容。

试题程序：

```
#include <stdio. h>

char xx[100] [11];
int yy[10];
int ReadDat (void);
void WriteDat(void);

void CountRs(void)
{

}

main ( )
{
int i;
for (i=0; i <10; i++)
yy[i] = 0;
if (ReadDat ())
{
```

```

printf ("选票数据文件 IN29.DAT 不能打开!\007\n");
return;
}
CountRs ();
WriteDat ();
}

int ReadDat (void)
{
FILE *fp;
int i;
char tt[13];
if((fp = fopen("IN29.DAT", "r")) == NULL)
return 1;
for (i = 0; i < 100; i++)
{
if(fgets(tt, 13, fp) == NULL)
return 1;
memcpy(xx[i], tt, 10);
}
fclose (fp);
return 0;
}
void WriteDat(void)
{
FILE *fp;
int i;
fp = fopen("OUT29.DAT", "w");
for(i = 0; i < 10; i++)
{
fprintf(fp, "%d\n", yy[i]);
printf("第%d 个人的选票数=%d\n", i+1, yy[i]);
}
fclose (fp);
}

```

35. 已知数据文件 IN77.DAT 中存有 200 个 4 位数，并已调用读函数 readDat()把这些数存入数组 a 中，请编制一函数 jsVal()，其功能是：如

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/655011202040011104>