

华中科技大学光学与电子信息学院考试试卷(B 卷)

2017 ~2018 学年度第 1 学期

课程名称: 软件技术基础

考试年级: 2017 级

考试时间: 2018 年 1 月 23 日

考试方式: 闭卷

学生姓名 学号 专业班级

题号	一	二	三	总分
题分	25	40	35	100
得分				

一、 单项选择 (25 分)

1. 对于下面的表示,请在相应小题前面的 () 中填入 A, B, C 或 D。

A, B, C, D 的含义如下:

(A) 合法的标识符; (B) C 语言的关键字; (C) 合法的字符串;

(D) 既不是 C 语言的关键字,也不是合法的标识符或字符串。

(C) (1) "I am OK!"

(B) (2) for

(A) (3) WHILE

(D) (4) How are you?

(C) (5) "Please input a:\n"

(A) (6) Window10

(A) (7) Main

(C) (8) "\123df\\n"

(A) (9) _good

(D) (10) 123OK

2. 对于下面的表示,请在相应小题前面的 () 中填入 A, B, C 或 D。

A, B, C, D 的含义如下:

(A) 合法的字符常数; (B) 合法的整常数; (C) 合法的浮点常数;

(D) 非法表示。

(C) (1) 1.24E-9

(A) (2) '\102'

(B) (3) 999L

(C) (4) 2.7F

(A) (5) '\X5F'

(D) (6) 0738

(D) (7) 56e0.5

(C) (8) 1e-2

(B) (9) 0xABCD

(A) (10) '\t'

3. 已知: `int a,b;` 用语句 `scanf(“%d%d”,&a,&b);` 输入 `a, b` 的值时, 不能作为数据输入的是 (A)

- (A) , (B) 空格 (C) 回车 (D) Tab 键

4. 下列程序执行后的输出结果是 (B)。

```
#include <stdio.h>

void main( )
{
    char ch = 'A' ;
    printf("%c\n", ch = ch>= 'A' &&ch<= 'Z' ?ch+ 'z' - 'Z' :ch);
}
```

- (A) A (B) a (C) Z (D) z

5. 下面一段程序段执行后的正确结果 (A)

```
int x=-1;
do
{
    x+=x*x;
}while (!x);
```

- (A) 是死循环 (B) 循环执行两次
(C) 循环执行一次 (D) 有语法错误

6. 以下叙述错误的是 (B)

- (A) `break` 语句不能用于循环语句和 `switch` 语句外的其他语句语句
(B) 在循环体内使用 `break` 语句或 `continue` 语句的作用相同
(C) 在循环语句中使用 `continue` 语句的作用是结束本次循环
(D) 在循环语句中使用 `break` 语句是为了跳出循环体, 提前结束循环

7. `char c[8] = "beijing", *s = c; int i;`则下面语句中，错误的是（ B ）

(A) `printf("%s\n", s);`

(B) `printf("%s\n", *s);`

(C) `for(i=1; i<7; i++) { printf("%c", c[i]); }`

(D) `for(i=1; i<7; i++) { printf("%c", s[i]); }`

8. 以下程序运行结果是（ D ）

```
void main()
```

```
{
```

```
    int x[]={1,2,4}, s, *p=x, i;
```

```
    for(i=0, s=1; i<=2; i++, p++)
```

```
        s*=*p;
```

```
    printf( "%d\n", s );
```

```
}
```

(A) 2 (B) 4 (C) 6 (D) 8

9. 已知：`char s[10], *s2="ab\0cdef";` 则执行语句 `strcpy(s1, s2);` 之后，

变量 `s1` 的内容为（ C ）

(A) `cdef` (B) `ab\0cdef` (C) `ab` (D) 内容不定

10. 已知：`char **s;` 正确的语句是（ B ）

(A) `s = " computer" ;` (B) `*s = " computer" ;`

(C) `**s=" computer" ;` (D) `*s = 'A' ;`

11. 已知：`char *aa[2]={ "abcd", " ABCD" };`则以下说法正确的是（ D ）。

(A) `aa` 数组元素的值分别是 `"abcd"` 和 `" ABCD"`

(B) `aa` 是指针变量，它指向含有两个数组元素的字符型一维数组

(C) `aa` 数组的两个元素分别存放的是含有 4 个字符的一维字符数组的首地址

(D) `aa` 数组的两个元素中各自存放了字符 `'a'` 和 `'A'` 的地址

12. 已知函数定义如下:

```
abcd( char *s1, char *s2)
{ while(*s2++ = *s1++); }
```

则函数 abcd 的功能是 (A)

- (A) 串复制 (B) 求串长 (C) 串比较 (D) 串反向

13. 运行下列程序, 输出结果是(B)

```
struct contry {
    int num;
    char name[20];
} x[5]={ {1, "China"}, {2, "USA"}, {3, "France"}, {4, "Englan"}, {5, "Spanish"} }
;void main( )
{ int i;
    for (i=3; i<5; i++)
        printf("%d%c", x[i].num, x[i].name[0]); }
```

- (A) 3F4E5S (B) 4E5S (C) F4E (D) c2U3F4E

14. 程序输出结果正确的是 (A)

```
#include<stdio.h>
void main( )
{
    int a[2][3] = {{1, 2, 3}, {4, 5, 6}};
    int m, *ptr;
    ptr=&a[0][0];
    m = (*ptr)*(*ptr+2)*(*ptr+4);
    printf( "%d\n", m);
}
```

- A) 15 B) 48 C) 24 D) 无定值

15. 执行语句 “for(i=0; i++<3;)” 后 i 的值是(B)

- (A) 2 (B) 3 (C) 4 (D) 5

16. 若有以下定义和说明：

```
#include<stdio.h>

struct std
{
    char num[6];
    char name[8];
    float mark[4];
}a[30];

FILE *fp;
```

设文件中以二进制形式存有 1 个班的学生数据，且已经正确打开，文件指针定位于文件开头。若要从文件中读出 30 个学生的数据放入 a 数组中，以下不能实现此功能的语句是（ D ）。

- (A) for(i=0;i<30;i++)
 fread(&a[i], sizeof(struct std), 1L, fp);
- (B) for(i=0;i<30;i++)
 fread(a+i, sizeof(struct std), 1L, fp);
- (C) fread(a, sizeof(struct std), 30L, fp);
- (D) for(i=0;i<30;i++)
 fread(a[i], sizeof(struct std), 1L, fp);

17. 下面对派生类的描述中，错误的是（D ）。

- (A) 一个派生类可以作为另外一个派生类的基类
- (B) 派生类至少有一个基类
- (C) 派生类的成员除了它自己的成员外，还包含了它的基类的成员
- (D) 派生类中继承的基类成员的访问权限到派生类中保持不变

二、 填空（40 分）

1. 阅读程序并写出程序输出结果

```
#include <stdio.h>
void main()
{
    unsigned int a=10, b=17, c=5, d=0;

    printf("f1=%d\n", d*=(a&b)+(c<<1|d));
    printf("f2=%d\n", !(a+b)+c-1||b+c/3);
    printf("f3=%d\n", a++-b+++c);
}
```

程序输出结果：

```
f1=0
f2=1
f3=-2
```

2. 补充程序语句，使其完成功能：多个字符串的排序。

```
#include <stdio.h>
#include <string.h>

void sortstr(char (*v)[20], int n)
{
    int i, j;
    char temp[20];

    for(i=0; i<n-1; i++)
    {
        for(j=i+1; j<n; j++)
        {
            if(strcmp(v[i], v[j])>=0)
            {
                strcpy(temp, v[i]);
                strcpy(v[i], v[j]);
                strcpy(v[j], temp);
            }
        }
    }
}

void main( )
{
    int i;
    char pronomes[20]={"pascal", "basic", "cobol", "prolog", "lisp"};

    sortstr(pronomes, 5);

    for(i=0; i<5; i++)
    {
        printf("%s\n", pronomes[i]);
    }
}
```

```
}
```

3. 阅读程序并写出程序输出结果

```
#include <stdio.h>
int n=1;
void func();

void main()
{
    static int x=5;
    int y;
    y=n;
    printf("main:x=%d, y=%d, n=%d\n", x, y, n);
    func();
    printf("main:x=%d, y=%d, n=%d\n", x, y, n);
}

void func()
{
    static int x=4;
    int y=10;
    x++;
    n++;
    printf("func:x=%d, y=%d, n=%d\n", x, y, n);
}
```

程序输出结果:

```
main:x=5, y=1, n=1
func :x=5, y=10, n=2
main:x=5, y=1, n=2
```

4. 阅读程序并写出程序输出结果

```
#include <stdio.h>
#define NULL 0
void main()
{
    char **pp;
    char *name[]={ "fortran", "basic", "pascal", "csharp", "c++" };
    pp=name;
    printf("%s\n", ++*++pp);
    printf("%s\n", *++pp +3);
    printf("%s\n", pp[2] +1);
}
```

程序输出结果:

```
asic
cal
++
```

5. 阅读程序并写出程序输出结果

```
#include<stdio.h>
struct Key
{
    char *keyword;
    int keyno;
};
void main( )
{
    struct Key  kd[3] = {"are",123}, {"my",456}, {"you", 789}};
    struct Key *p;
    p = kd;
    printf("keyno=%d\n", ++p->keyno);
    printf("keyno=%d\n", (++p)->keyno);
    p = kd;
    printf("keyword=%s\n", p->keyword);
}
```

程序输出结果:

```
keyno=124
keyno=456
keyword=are
```

6. 阅读程序并写出程序输出结果

```
#include <string.h>
#include <iostream.h>

class Student
{
private:
    int    num;
    char   name[20];
    int    score;
public:
    Student(int a, char *b, int c )
```



```

{
    num = a;
    strcpy(name,b);
    score =c;
}

~Student()
{
    cout<<"num:"<<num;
    cout<<" name:"<<name;
    cout<<" score:"<<score<<endl;
}

void display()
{
    cout<<"num:"<<num;
    cout<<" name:"<<name;
    cout<<" score:"<<score<<endl;
}
};

class Student1 : public Student
{
private:
    int age;
    char addr[20];

public:
    Student1(int x, char *y, int z, int a, char *b)
        :Student(x,y,z)
    {
        age = a;
        strcpy(addr,b);
    }

    ~Student1()
    {
        cout<<"age:"<<age;
        cout<<" address:"<<addr<<endl;
    }

    void display_1()
    {
        cout<<"age:"<<age;

```

```
        cout<<" address:"<<addr<<endl;
    }
};

int main( )
{
    Student1 stu(2017001,"liu",90,18,"HUST N5_101");
    stu.display();
    return 0;
}
```

程序输出结果:

```
num:2017001 name:liu score:90
age:18 address:HUST N5 101
num:2017001 name:liu score:90
```



```

        c[k++]=a[i];                                //1 分

for(i=0;i<j-1;i++) //对奇数按从小到大排序 2 分
{
    for(m=i+1;m<j;m++)
        if(b[i]>b[m])
        {
            t=b[i];
            b[i]=b[m];
            b[m]=t;
        }
}
for(i=0;i<k-1;i++) //对偶数从大到小排序 2 分
{
    for(m=i+1;m<k;m++)
        if(c[i]<c[m])
        {
            t=c[i];
            c[i]=c[m];
            c[m]=t;
        }
}
for(i=0;i<j;i++) //分别输出排序后的奇数和偶数数组 2 分
    printf("%d\n",b[i]);
for(i=0;i<k;i++)
    printf("%d\n",c[i]);

}

```

第二种答案：用原数组排序

```
#include <stdio.h>
```

```
void main() // 头部分 1 分
```

```

{
    int a[10]={2, 17, 66, 14, 5, 22, 37, 55, 8, 99}; //定义数组并初始化 1 分
    int i, j, k, m, t;
    for(i=0, j=0; i<10; i++) //判读奇数个数，并将奇数全部放到原数组前段
        if(a[i]%2!=0)
        {
            t=a[i];
            a[i]=a[j];
            a[j]=t;
            j++;
        } //2 分
}

```

```

for(i=0;i<j-1;i++) //对奇数按从小到大排序 2 分
{
    for(m=i+1;m<j;m++)
        if(a[i]>a[m])
        {
            t=a[i];
            a[i]=a[m];
            a[m]=t;
        }
}
for(i=j;i<9;i++) //对偶数从大到小排序 2 分
{
    for(m=i+1;m<10;m++)
        if(a[i]<a[m])
        {
            t=a[i];
            a[i]=a[m];
            a[m]=t;
        }
}
for(i=0;i<10;i++) //输出排序后的数组 1 分
    printf("%d\n",a[i]);
}

```

3. 输入一行英文，输出英文中最大的单词（按字典序）。(9 分)

```

#include<stdio.h>

#include<string.h>

#define MAXLENGTH 100

#define MAXWORDLENGTH 20

void FindMaxString(char *str, char *max);

void main(void) // 头及函数声明1分
{
    char str[MAXLENGTH],max[MAXWORDLENGTH] = ""; // 数据定义1分
    printf("Please input the string : ");
    gets(str); // 数据输入1分
    FindMaxString(str, max); // 函数调用1分
    printf("The max word is : ");
}

```

```

puts(max); // 数据输出1分
}
void FindMaxString(char *str, char *max)// 数据定义1分
{
    char word[MAXWORDLENGTH];
    char *p = word;
    while(*str != '\0') //算法部分3分
    {
        for(;*str == ' '; str++);
        for(;*str != ' ' && *str != '\0';)
            *p++ = *str++;
        *p = '\0';
        if(strcmp(p = word, max) > 0)
            strcpy(max, word);
    }
}

```

4. 有 10 个考生，考生信息包括编号、姓名、性别和 4 门功课的成绩，main 函数负责输入考生信息，并统计每个学生的总分。编写函数实现：（1）将所有考生按总分进行排序；（2）找出总分最高的男考生信息并输出；（3）找出平均分不及格（<60 分）的女考生信息并输出。

```

#include <stdio.h>
#define N 10
struct
{
    long ID;
    char name[20];
    char sex;
    float score[4];
}

```

```

    float total;
} STUDENT;

void sort(STUDENT data[ ],int n);
void search_boy(STUDENT data[ ],int n);
void search_girl(STUDENT data[ ],int n);
//数据定义及函数声明 2 分
void main( )
{
    STUDENT stu[N];

    int i,j;
    printf("录入并统计每个学生的总分并记录:\n");
    for(i=0;i<N;i++)
    {
        printf("No%d:\n",i+1);
        scanf("%d",&stu[i].ID);
        scanf("%s",stu[i].name);
        scanf("%c",&stu[i].sex);
        stu[i].total=0.0;
        for(j=0;j<4;j++)
        {
            int abc;
            scanf("%f",&abc);
            stu[i].total +=(stu[i].score[j]=abc);
        }
    } //数据输入及统计 2 分

    sort(stu,N);
    search_boy(stu,N);
    search_girl(stu,N); //函数调用 1 分
}

```

```

void sort(STUDENT data[ ], int n)//排序算法 2 分
{
    int i, j;
    for(i=1; i<=n-1; i++)
        for(j=1; j<=n-i; j++)
            if(data[j-1].total < data[j].total)
            {
                STUDENT temp;
                temp =data[j];
                data[j] =data[j-1];
                data[j-1]=temp;
            }
}

```

```

void search_boy(STUDENT data[ ], int n)//1 分
{
    for(i=0; i<n; i++)
        if(data.sex==1)
        {
            printf("%d %s %d\n", stu[i].ID, stu[i].name,
                stu[i].total);
            return;
        }
}

```

```

void search_girl(STUDENT stu[ ], int n)//1 分
{
    int i;

```



```
for(i=0;i<n;i++)
{
    if(stu[i].sex==0&&stu[i].total/4<60)
        printf("%d %s %d\n",stu[i].ID,stu[i].name,
            stu[i].total);
}
}
```