

第三部分

综合训练

一、单选题

1. C 语言程序的基本单位是()。
A. 过程 B. 函数 C. 子程序 D. 标识符
2. 下列字符序列中,可用作 C 标识符的一组字符序列是()。
A. S, b, sum, average, _above B. class, day, lotus_1, 2day
C. #md, &12x, month, student_n! D. D56, r_1_2, name, _st_1
3. 以下标识符中,不能作为合法的 C 用户定义标识符的是()。
A. a3_b3 B. void C. _123 D. IF
4. 下列变量说明语句中,正确的是()。
A. char: a b c; B. char a; b; c; C. int x; z; D. int x, z;
5. 执行下列语句后变量 x 和 y 的值是()。

```
y = 10; x = y++;
```


A. x=10, y=10 B. x=11, y=11 C. x=10, y=11 D. x=11, y=10
6. 设 a=12,且 a 定义为整型变量。执行语句“a+=a-=a*=a;”后 a 的值为()。
A. 12 B. 144 C. 0 D. 132
7. 为了提高程序的运行速度,在函数中对于整型或指针可以使用()型的变量。
A. auto B. register C. static D. extern
8. 以下程序的输出结果为()。

```
void main()  
{ int i = 010, j = 10; printf("%d, %d\n", ++i, j--); }
```


A. 11, 10 B. 9, 10 C. 010, 9 D. 10, 9
9. 在 C 语言中,if 语句后的一对圆括号中,用以决定分支流程的表达式是()。
A. 只能用逻辑表达式 B. 只能用关系表达式
C. 只能用逻辑表达式或关系表达式 D. 可用任意表达式
10. 在以下表达式中,与 do...while(E) 语句中的(E)不等价的表达式是()。
A. (! E==0) B. (E>0||E<0)
C. (E==0) D. (E!=0)
11. C 语言的 if 语句嵌套时,if 与 else 的配对关系是()。
A. 每个 else 总是与它上面的最近的 if 配对

- B. 每个 else 总是与最外层的 if 配对
C. 每个 else 与 if 的配对是任意的
D. 每个 else 总是与它上面的 if 配对
12. 下列数组说明中,正确的是()。
- A. static char str[]="China";
B. static char str[]; str="China";
C. static char str1[5],str2[]={ "China" }; str1=str2;
D. static char str1[],str2[];str2={ "China" }; strcpy(str1,str2);
13. 下列定义数组的语句中正确的是()。
- A. #define size 10 char str1[size],str2[size+2];
B. char str[];
C. int num['10'];
D. int n=5; int a[n][n+2];
14. 下列定义数组的语句中不正确的是()。
- A. static int a[2][3]={1,2,3,4,5,6};
B. static int a[2][3]={ {1},{4,5} };
C. static int a[][3]={ {1},{4} };
D. static int a[][]={ {1,2,3},{4,5,6} };
15. 下列语句中,不正确的是()。
- A. static char a[2]={1,2};
B. static int a[2]={ '1','2' };
C. static char a[2]={ '1','2','3' };
D. static char a[2]={ '1' };
16. 在 C 语言程序中()。
- A. 函数的定义可以嵌套,但函数的调用不可以嵌套
B. 函数的定义不可以嵌套,但函数的调用可以嵌套
C. 函数的定义和函数调用均可以嵌套
D. 函数的定义和函数调用不可以嵌套
17. 当数组名作为实参传递给函数时,数组名被处理为()。
- A. 该数组的长度
B. 该数组的元素个数
C. 该数组的首地址
D. 该数组中各元素的值
18. 以下函数调用语句中实参的个数是()。

```
exce((v1,v2),(v3,v4,v5),v6);
```

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

19. 下列程序的运行结果为()

```
#define P 3  
#define S(a) P*a*a  
voidmain()
```

```

{   int   ar;
    ar = S(3 + 5);
    printf("\n %d", ar);
}

```

- A. 192 B. 29 C. 27 D. 25
20. 已知 p、p1 为指针变量, a 为数组名, j 为整型变量, 下列赋值语句中不正确的是()。
- A. p=&j, p=p1; B. p=a; C. p=&a[j]; D. p=10;
21. 两个指针变量不可以()。
- A. 相加 B. 比较 C. 相减 D. 指向同一地址
22. 若已定义 x 为 int 类型变量, 则下列语句中说明指针变量 p 的正确语句是()。
- A. int p=&x; B. int *p=x; C. int *p=&x; D. *p=*x;
23. 若有定义“char *p1, *p2, *p3, *p4, ch;”, 则不能正确赋值的程序语句为()。
- A. p1=&ch; scanf("%c", p1);
- B. p2=(char *)malloc(1); scanf("%c", p2);
- C. *p3=getchar();
- D. p4=&ch; *p4=getchar();
24. 在位运算中, 操作数每右移一位, 其结果相当于()。
- A. 操作数乘以 2 B. 操作数除以 2
- C. 操作数乘以 16 D. 操作数除以 16
25. C 语言中文件的类型只有()。
- A. 索引文件和文本文件两种 B. ASCII 文件和二进制文件两种
- C. 文本文件一种 D. 二进制文件一种
26. C 语言中文件的存储方式是()。
- A. 只能顺序存取 B. 只能随机存取(或直接存取)
- C. 可以顺序存取, 也可随机存取 D. 只能从文件的开头进行存取
27. 打开文件时, 方式“w”决定了对文件进行的操作是()。
- A. 只写盘 B. 只读盘 C. 可读可写盘 D. 追加写盘
28. 若有 int a[10]={1, 2, 3, 4, 5, 6, 7, 8}; int *p; p=&a[5]; p[-3] 的值是()。
- A. 2 B. 3 C. 4 D. 不一定
29. C 语言规定, 函数返回值的类型是由()。
- A. return 语句中的表达式类型所决定
- B. 调用该函数时的主调函数类型所决定
- C. 调用该函数时系统临时决定
- D. 定义该函数时所指定的函数类型所决定
30. 关于建立函数的目的, 以下正确的说法是()。
- A. 提高程序的执行效率 B. 提高程序的可读性
- C. 减少程序的篇幅 D. 减少程序文件所占内存

二、程序改错

1. 功能: 在一个已按升序排列的数组中插入一个数, 插入后数组元素仍按升序排列。

```
# define N 11
void main()
{
    int i,number,a[N] = {1,2,4,6,8,9,12,15,149,156};
    printf("please enter an integer to insert in the array:\n");
    / ***** FOUND ***** /
    scanf("%d",&number)
    printf("The original array:\n");
    for(i = 0;i < N - 1;i++)
        printf("%5d",a[i]);
    printf("\n");
    / ***** FOUND ***** /
    for(i = N - 1;i >= 0;i--)
        if(number <= a[i])
            / ***** FOUND ***** /
            a[i] = a[i - 1];
        else
        {
            a[i + 1] = number;
            / ***** FOUND ***** /
            exit;
        }
    if(number < a[0]) a[0] = number;
    printf("The result array:\n");
    for(i = 0;i < N;i++)
        printf("%5d",a[i]);
    printf("\n");
}
```

2. 功能：判断 m 是否为素数，若是返回 1；否则返回 0。

```
/ ***** FOUND ***** /
void fun( int n)
{
    int i,k = 1;
    if(m <= 1) k = 0;
    / ***** FOUND ***** /
    for(i = 1;i < m;i++)
        / ***** FOUND ***** /
        if(m % i == 0) k = 0;
    / ***** FOUND ***** /
    return m;
}
void main()
{
    int m,k = 0;
    for(m = 1;m < 100;m++)
        if(fun(m) == 1)
        {
            printf("%4d",m);k++;
            if(k % 5 == 0) printf("\n");
        }
}
```

3. 功能：用下面的和式求圆周率的近似值，直到最后一项的绝对值小于等于 0.0001。

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots$$

```

/ ***** FOUND ***** /
#include <stdlib.h>
fun()
{
    int i = 1;
    / ***** FOUND ***** /
    int s = 0, t = 1, p = 1;
    / ***** FOUND ***** /
    while(fabs(t) <= 1e-4)
    {
        s = s + t;
        p = -p;
        i = i + 2;
        t = p/i;
    }
    / ***** FOUND ***** /
    printf("pi = %d\n", s * 4);
}
void main()
{
    fun();
}

```

4. 功能：计算并输出 k 以内最大的 10 个能被 13 或 17 整除的自然数之和。k 的值由主函数输入。例如，若 k 的值为 500，则函数值为 4622。

```

int fun(int k)
{
    int m = 0, mc = 0;
    / ***** FOUND ***** /
    while ((k >= 2) || (mc < 10))
    {
        / ***** FOUND ***** /
        if ((k % 13 == 0) || (k % 17 == 0))
        {
            m = m + k;
            mc++;
        }
        / ***** FOUND ***** /
        k++;
    }
    / ***** FOUND ***** /
    return;
}
void main()
{
    printf("%d\n", fun(500));
}

```

5. 功能：求出两个非零正整数的最大公约数，并作为函数值返回。例如，若给 num1 和 num2 分别输入 49 和 21，则输出的最大公约数为 7。

```

int fun(int a, int b)

```

```
{    int r,t;
    if(a<b)
    {    t=a;
        / ***** FOUND ***** /
        b=a;
        / ***** FOUND ***** /
        a=t;
    }
    r=a%b;
    while(r!=0)
    {    a=b;
        b=r;
        / ***** FOUND ***** /
        r=a/b;
    }
    / ***** FOUND ***** /
    return a;
}

void main()
{    int num1,num2,a;
    scanf("%d%d",&num1,&num2);
    a=fun(num1,num2);
    printf("the maximum common divisor is %d\n\n",a);
}
```

6. 功能：计算数组元素中值为正数的平均值(不包括0)。

```
double fun(int s[])
{    / ***** FOUND ***** /
    int sum=0.0;
    int c=0,i=0;
    / ***** FOUND ***** /
    while(s[i]!=0)
    {    if (s[i]>0)
        {    sum+=s[i];
            c++;
        }
        i++;
    }
    / ***** FOUND ***** /
    sum/=c;
    / ***** FOUND ***** /
    return c;
}

void main()
{
    int x[1000];int i=0;
    do{    scanf("%d",&x[i]);    } while(x[i++]!=0);
    printf("%f\n",fun(x));
}
```

7. 功能: 找出大于 m 的最小素数, 并将其作为函数值返回。

```
int fun(int m)
{   int i,k;
    for(i = m + 1;;i++)
    {   / ***** FOUND ***** /
        for(k = 1;k < i;k++)
            / ***** FOUND ***** /
            if(i % k != 0) break;
        / ***** FOUND ***** /
        if(k < i)
            / ***** FOUND ***** /
            return k;
    }
}

void main()
{   int n;
    scanf(" %d",&n);
    printf(" %d\n",fun(n));
}
```

8. 功能: 实现两个字符串的连接。

```
void main()
{   char s1[80],s2[80];
    void scat(char s1[],char s2[]);
    gets(s1);
    gets(s2);
    scat(s1,s2);
    puts(s1);
}

void scat(char s1[],char s2[])
{   int i = 0,j = 0;
    / ***** FOUND ***** /
    while(s1[i] != '\0')
        i++;
    / ***** FOUND ***** /
    while(s2[j] != '\0')
    {   / ***** FOUND ***** /
        s2[j] = s1[i];
        i++;
        j++;
    }
    / ***** FOUND ***** /
    s2[j] = '\0';
}
```

9. 功能: 有一数组内放 10 个整数, 要求找出最小数和它的下标, 然后把它和数组中最前面的元素即第一个数对换位置。

```
void main()
{   int i,a[10],min,k = 0;
```

```

printf("\n please input array 10 elements\n");
for(i = 0; i < 10; i++)
/ ***** FOUND ***** /
    scanf(" %d", a[i]);
for(i = 0; i < 10; i++)
    printf(" %d", a[i]);
min = a[0];
/ ***** FOUND ***** /
for(i = 3; i < 10; i++)
/ ***** FOUND ***** /
if(a[i] > min)
{
    min = a[i];
    k = i;
}
/ ***** FOUND ***** /
a[k] = a[i];
a[0] = min;
printf("\n after eschange:\n");
for(i = 0; i < 10; i++)
    printf(" %d", a[i]);
printf("\nk = %d\nmin = %d\n", k, min);
}

```

10. 功能：先将在字符串 s 中的字符按正序存放到字符串 t 中，然后把 s 中的字符按逆序连接到字符串 t 的后面。

```

void fun(char * s, char * t)
{
    int i, sl;
    sl = strlen(s);
/ ***** FOUND ***** /
    for(i = 0; i < sl; i++)
        t[i] = s[i];
    for(i = 0; i < sl; i++)
/ ***** FOUND ***** /
        t[sl + i] = s[sl - i];
/ ***** FOUND ***** /
        t[sl] = '\0';
}
void main()
{
    char s[100], t[100];
    printf("\nPlease enter string s:"); scanf(" %s", s);
    fun(s, t);
    printf("The result is: %s\n", t);
}

```

11. 功能：求出数组中最大数和次最大数，并把最大数和 a[0] 中的数对调、次最大数和 a[1] 中的数对调。

```

#define N 20
void fun ( int * a, int n )
{
    int i, m, t, k ;

```



```

for(i = 0; i < 2; i++)
{
    / ***** FOUND ***** /
    m = 0;
    / ***** FOUND ***** /
    for(k = 1; k < n; k++)
        / ***** FOUND ***** /
        if(a[k] > a[m]) k = m;
        t = a[i]; a[i] = a[m]; a[m] = t;
}
}
void main()
{
    int b[N] = {11, 5, 12, 0, 3, 6, 9, 7, 10, 8}, n = 10, i;
    for( i = 0; i < n; i++) printf(" %d ", b[i]);
    printf("\n");
    fun( b, n );
    for( i = 0; i < n; i++) printf(" %d ", b[i]);
    printf("\n");
}

```

12. 功能：写一个函数，求一个字符串的长度，在 main() 函数中输入字符串，并输出其长度。

```

int length( char *p)
{
    int n; n = 0;
    / ***** FOUND ***** /
    while( *p == '\0')
    {
        n++;
        p++;
    }
    return n;
}
void main()
{
    int len;
    / ***** FOUND ***** /
    char *str[20];
    printf("please input a string:\n");
    scanf(" %s", str);
    / ***** FOUND ***** /
    len == length(str);
    printf("the string has %d characters.", len);
}

```

三、程序填空

1. 功能：求 $1! + 3! + 5! + \dots + n!$ 的和。

```

void main()
{
    long int f, s;
    int i, j, n;
    / ***** SPACE ***** /
    【?】;
    scanf(" %d", &n);
}

```

```

/ ***** SPACE ***** /
for(i = 1; i <= n; 【?】)
{
    f = 1;
    / ***** SPACE ***** /
    for(j = 1; 【?】; j++)
        / ***** SPACE ***** /
        【?】;
    s = s + f;
}
printf("n = %d, s = %ld\n", n, s);
}

```

2. 功能：求 100~999 之间的水仙花数。

```

int fun(int n)
{
    int i, j, k, m;
    m = n;
    / ***** SPACE ***** /
    【?】;
    for(i = 1; i < 4; i++)
    {
        / ***** SPACE ***** /
        【?】;
        m = (m - j) / 10;
        k = k + j * j * j;
    }
    if(k == n)
        / ***** SPACE ***** /
        【?】;
    else
        return(0);
}

void main()
{
    int i;
    for(i = 100; i < 1000; i++)
        / ***** SPACE ***** /
        if(【?】 == 1)
            printf("%d is ok!\n", i);
}

```

3. 功能：产生并输出杨辉三角的前 7 行。

```

void main()
{
    int a[7][7], i, j;
    for(i = 0; i < 7; i++)
    {
        a[i][0] = 1;
        / ***** SPACE ***** /
        【?】;
    }
    for(i = 2; i < 7; i++)
        / ***** SPACE ***** /
        for (j = 1; j < 【?】; j++)

```