

1. (A) 是构成 C 语言程序的基本单位。

A、函数 B、过程 C、子程序 D、子例程

2. C 语言程序从 C 开始执行。

A、程序中第一条可执行语句 B、程序中第一个函数
C、程序中的 main 函数 D、包含文件中的第一个函数

3. 以下说法中正确的是 (C)。

A、C 语言程序总是从第一个定义的函数开始执行
B、在 C 语言程序中，要调用的函数必须在 main() 函数中定义
C、C 语言程序总是从 main() 函数开始执行
D、C 语言程序中的 main() 函数必须放在程序的开始部分

4. 下列关于 C 语言的说法错误的是 (B)。

A、C 程序的工作过程是编辑、编译、连接、运行 B、C 语言不区分大小写。
C、C 程序的三种基本结构是顺序、选择、循环 D、C 程序从 main 函数开始执行

5. 下列正确的标识符是 (C)。

A、-a1 B、a[i] C、a2_i D、int t

5~8 题为相同类型题

考点：标识符的命名规则只能由字母、数字、下划线构成，数字不能作为标识符的开头，关键字不能作为标识符。选项 A 中的“-”，选项 B 中“[”与“]”不满足 (1)；选项 D 中的 int 为关键字，不满足 (3)

6. 下列 C 语言用户标识符中合法的是 (B)。

A、3ax B、x C、case D、-e2 E)union

选项 A 中的标识符以数字开头不满足 (2)；选项 C、E 均为为关键字，不满足 (3)；选项 D 中的“-”不满足 (1)；

7. 下列四组选项中，正确的 C 语言标识符是 (C)。

A、%x B、a+b C、a123 D、123

选项 A 中的“%”，选项 B 中“+”不满足 (1)；选项 D 中的标识符以数字开头不满足 (2)

8. 下列四组字符串中都可以用作 C 语言程序中的标识符的是 (A)。

A、print _3d db8 aBc B、I\am one_half start\$it 3pai
C、str_1 Cpp pow while D、Pxq My->book line# His.age

选项 B 中的“\”，“\$”，选项 D 中“>”，“#”，“.”，“-”不满足 (1)；选项 C 中的 while 为关键字，不满足 (3)

9. C 语言中的简单数据类型包括 (D)。

A、整型、实型、逻辑型 B、整型、实型、逻辑型、字符型
C、整型、字符型、逻辑型 D、整型、实型、字符型

10. 在 C 语言程序中，表达式 5%2 的结果是 C。

A、2.5 B、2 C、1 D、3

11. C 语言中，关系表达式和逻辑表达式的值是 (B)。

A、0 B、0 或 1 C、1 D、‘T’ 或 ‘F’

12. 下面 (D) 表达式的值为 4。

A、11/3 B、11.0/3 C、(float)11/3 D、(int)(11.0/3+0.5)

选项 A，11 与 3 为两个整数，11/3 结果的数据类型也应为整数，因此将 3.666666 的小数

部分全部舍掉，仅保留整数，因此 $11/3=3$ 。

选项 B，11.0 为实数，3 为整数，因此首先要统一数据类型，将整型数据 3 转换为 3.0，转换后数据类型统一为实型数据，选项 B 变为 $11.0/3.0$ ，结果的数据类型也应为实型数据，因此选项 B $11.0/3=3.666666$

选项 C，先将整数 11 强制类型转换，转换为实型 11.0，因此选项 C 变为 $11.0/3$ ，其后计算过程、结果与选项 B 同

选项 D，首先计算 $11.0/3$ ，其计算过程、结果与选项 B 同，得到 3.666666；再计算 $3.666666+0.5=4.166666$ ，最后将 4.166666 强制类型转换为整型，即将其小数部分全部舍掉，结果为 4

13. 设整型变量 $a=2$ ，则执行下列语句后，浮点型变量 b 的值不为 0.5 的是 (B)

A、 $b=1.0/a$ B、 $b=(float)(1/a)$ C、 $b=1/(float)a$ D、 $b=1/(a*1.0)$

14. 若 “ $\text{int } n; \text{float } f=13.8;$ ”，则执行 “ $n=(\text{int})f\%3$ ” 后， n 的值是 (A)

A、1 B、4 C、4.333333 D、4.6

“(int)f” 表示将 f 中的值强制类型转换为整型，即将 13.8 的小数部分舍掉，转换为 13；然后计算 $13\%3$ ，结果为 1，再将结果赋给变量 n ，因此 n 的值为 1

15. 以下对一维数组 a 的正确说明是： D

A、 $\text{char } a(10);$ B、 $\text{int } a[];$
C、 $\text{int } k=5, a[k];$ D、 $\text{char } a[3]=\{ 'a', 'b', 'c' \};$
一维数组的定义、初始化 类型符 数组名 [常量表达式]

类型符是指数组中数组元素的类型；数组名要符合标识符命名规则；常量表达式是指数组的长度（数组中包含元素的个数），其值只能是整数，不可以是变量，而且从 1 开始计数。选项 A，常量表达式只能放在中括号 $[]$ 中。选项 B，只有在对数组初始化（即赋值）的时候才可以省略数组的长度，B 中并未对 a 进行初始化。选项 C，常量表达式不能为变量。

16. 以下能对一维数组 a 进行初始化的语句是：(C)

A、 $\text{int } a[5]=(0,1,2,3,4,)$ B、 $\text{inta}(5)={}$
C、 $\text{int } a[3]={0,1,2}$ D、 $\text{int } a\{5\}=\{10*1\}$

选项 B、D，常量表达式只能放在中括号 $[]$ 中。选项 A，数组可以看做是若干个相同数据类型元素的有序集合，因此以集合的形式对其初始化，使用 $\{ \}$ 对其初始化，选项 A 用了 $()$ 。

17. 在 C 语言中对一维整型数组的正确定义为 D。

A、 $\text{int } a(10);$ B、 $\text{int } n=10, a[n];$
C、 $\text{int } n; a[n];$ D、 $\#define N 10$
 $\text{int } a[N];$

18. 已知： $\text{int } a[10];$ 则对 a 数组元素的正确引用是 (D)。

A、 $a[10]$ B、 $a[3.5]$ C、 $a(5)$ D、 $a[0]$

数组元素的引用 数组名 [下标]

引用数组元素时， $[]$ 中的下标为逻辑地址下标，只能为整数，可以为变量，且从 0 开始计数。 $\text{int } a[10]$ 表示定义了一个包含 10 个整型数据的数组 a ，数组元素的逻辑地址下标范围为 $0\sim9$ ，即 $a[0]$ 表示组中第 1 个元素； $a[1]$ 表示组中第 2 个元素； $a[2]$ 表示组中第 3 个元素；……； $a[9]$ 表示组中第 10 个元素。选项 A，超过了数组 a 的逻辑地址下标范围；选项 B，逻辑地址下标只能为整数。选项 C，逻辑地址下标只能放在 $[]$ 中

19. 若有以下数组说明，则 $i=10; a[a[i]]$ 元素数值是 (C)。

$\text{int } a[12]=\{1, 4, 7, 10, 2, 5, 8, 11, 3, 6, 9, 12\};$
A、10 B、9 C、6 D、5

20. 若有说明: `int a[][3]={1, 2, 3}, {4, 5}, {6, 7}`; 则数组 `a` 的第一维的大小为: (B)

A、2 B、3 C、4 D、无确定值

二维数组的一维大小, 即指二维数组的行数, 在本题中, 按行对二维数组赋值, 因此内层有几个大括号, 数组就有几行

21. 对二维数组的正确定义是 (C)

A、`int a[][]={1, 2, 3, 4, 5, 6}`; B、`int a[2][]={1, 2, 3, 4, 5, 6}`;

C、`int a[][3]={1, 2, 3, 4, 5, 6}`; D、`int a[2, 3]={1, 2, 3, 4, 5, 6}`;

二维数组的定义、初始化 类型符 数组名 [常量表达式][常量表达式]

二维数组可以看做是矩阵. 类型符是指数组中数组元素的类型; 数组名要符合标识符命名规则; 第一个常量表达式是指数组的行数; 第二个常量表达式是指数组的列数; 常量表达式的值只能是整数, 不可以是变量, 而且从 1 开始计数。

一维数组初始化时可以省略数组长度. 二维数组初始化时可以省略行数, 但不能省略列数

选项 A, B, 都省略了列数. 选项 D, 不符合二维数组定义的一般形式, 行、列常量表达式应该放在不同的 [] 中

22. 已知 `int a[3][4]`; 则对数组元素引用正确的是 C

A、`a[2][4]` B、`a[1, 3]` C、`a[2][0]` D、`a(2)(1)`

数组元素的引用 数组名[下标][下标]

引用数组元素时, [] 中的下标为逻辑地址下标, 只能为整数, 可以为变量, 且从 0 开始计数. 第一个[下标]表示行逻辑地址下标, 第二个[下标]表示列逻辑地址下标. 因此 `a` 的行逻辑地址范围 $0 \sim 2$; `a` 的列逻辑地址范围 $0 \sim 3$; 选项 A, 列逻辑地址下标超过范围, 选项 B, D, 的引用形式不正确。

23. C 语言中函数返回值的类型是由 A 决定的。

A、函数定义时指定的类型 B、return 语句中的表达式类型

C、调用该函数时的实参的数据类型 D、形参的数据类型

24. 在 C 语言中, 函数的数据类型是指 (A)

A、函数返回值的类型 B、函数形参的数据类型

C、调用该函数时的实参的数据类型 D、任意指定的数据类型

25. 在函数调用时, 以下说法正确的是 (B)

A、函数调用后必须带回返回值 B、实际参数和形式参数可以同名

C、函数间的数据传递不可以使用全局变量 D、主调函数和被调函数总是在同一个文件里

26. 在 C 语言中, 表示静态存储类别的关键字是: (C)

A、auto B、register C、static D、extern

27. 未指定存储类别的变量, 其隐含的存储类别为 (A)。

A、auto B、static C、extern D、register

28. 若有以下说明语句:

```
struct student
{
    int num;
    char name[ ];
    float score;
}stu;
```

则下面的叙述不正确的是: (D)

A、struct 是结构体类型的关键字

B、struct student 是用户定义的结构体类型

C、num, score 都是结构体成员名

D、stu 是用户定义的结构体类型名

29. 若有以下说明语句:

```
struct date
```

```
{ int year;
```

```
int month;
```

```
int day;
```

```
}brithday;
```

则下面的叙述不正确的是 C .

A、struct 是声明结构体类型时用的关键字

B、struct date 是用户定义的结构体类型名

C、brithday 是用户定义的结构体类型名

D、year, day 都是结构体成员名

30. 以下对结构变量 stu1 中成员 age 的非法引用是 B

```
struct student
```

```
{ int age;
```

```
int num;
```

```
}stu1,*p;
```

```
p=&stu1;
```

A、stu1.age

B、student.age

C、p->age

D、(*p).age

31. 设有如下定义:

```
struck sk
```

```
{ int a;
```

```
float b;
```

```
}data;
```

```
int *p;
```

若要使 P 指向 data 中的 a 域, 正确的赋值语句是 C

A、p=&a;

B、p=datA、a;

C、p=&datA、a;

D、*p=datA、a;

32. 设有以下说明语句:

```
typedef struct stu
```

```
{ int a;
```

```
float b;
```

```
} stutype;
```

则下面叙述中错误的是 (D)。

A、struct 是结构类型的关键字

B、struct stu 是用户定义的结构类型

C、a 和 b 都是结构成员名

D、stutype 是用户定义的结构体变量名

33. 语句 int *p; 说明了 C 。

A、p 是指向一维数组的指针

B、p 是指向函数的指针, 该函数返回一 int 型数据

C、p 是指向 int 型数据的指针 // 指针的定义教材 P223

D、p 是函数名, 该函数返回一指向 int 型数据的指针

34. 下列不正确的定义是 (A)。

A、int *p=&i,i;

B、int *p,i;

C、int i,*p=&i;

D、int i,*p;

选项 A 先定义一个整型指针变量 p, 然后将变量 i 的地址赋给 p。然而此时还未定义变量 i 因此编译器无法获得变量 i 的地址。(A 与 C 对比, 选项 C 先定义变量 i, 则在内存中为 i 分配空间, 因此 i 在内存空间的地址就可以确定了; 然后再定义 p, 此时可以为 p 赋 i 的地址, C 正确)

35. 若有说明: int n=2,*p=&n,*q=p, 则以下非法的赋值语句是: (D)

A、p=q

B、*p=*q

C、n=*q

D、p=n

p, q 同为整型指针变量, 二者里面仅能存放整型变量的地址。

选项 A, q 中为地址, 因此可将此地址赋给 p

选项 B, *p 表示 p 所指向对象 n 的内容, 即一个整数; *q 表示 q 所指向对象的内容, 由于在定义 q 时为其初始化, 将 p 中 n 的地址给 q, 因此 p 中存放 n 的地址, *q 表示 q 所指向对象 n 的内容. 因此 *p=*q 相当于 n=n;

选项 C, n=*q 等价于 n=n;

选项 D, p 中只能存放地址, 不能将 n 中的整数值赋给 p

36. 有语句: int a[10]; 则 B 是对指针变量 p 的正确定义和初始化。

A、int p=*a; B、int *p=a; C、int p=&a; D、int *p=&a;

选项 A, a 是数组名, 不是指针变量名, 因此不可用 * 标注数组名 a 选项 C, a 是数组名, 数组名就是地址, 无需再用地址符号。而且在定义指针变量 p 时, 应在变量名前加 *, 表明 p 是指针变量 选项 D, a 是数组名, 数组名就是地址, 无需再用地址符号。

37. 若有说明语句 “int a[5], *p=a;”, 则对数组元素的正确引用是 (C)。

A、a[p] B、p[a] C、*(p+2) D、p+2

首先定义一个整型数组 a, a 的长度为 5, 然后定义一个指针变量 p, 同时对 p 进行初始化, 将数组 a 的地址赋给 p。因此此时 p 中存放的数组 a 的首地址, 即数组中第一个元素 a[0] 的地址。

对于数组元素下标的引用, 一般形式 数组名[下标] 其中下标为逻辑地址下标, 从 0 开始计数, 方括号中的下标可以是变量, 可以是表达式, 但结果一定要是整数。

选项 A, p 中存放的是地址, 不是整数, 不能做数组元素的下标

选项 B, a 是数组名, 数组名就是地址, 不是整数, 不能做数组元素的下标

选项 C, (重点!!! 详见 p231~234) p+2 表示指向同一数组中的下两个元素的地址, 当前 p 指向 a[0], 则 p+2 表示 a[2] 的地址, 因此 *(p+2) 表示 a[2] 的内容

38. 有如下程序

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 B

A、*P+9 B、*(P+8) C、*P+=9 D、P+8

首先定义一个整型数组 a, a 的长度为 5, 然后定义一个指针变量 P, 同时对 P 进行初始化, 将数组 a 的地址赋给 P。因此此时 P 中存放的数组 a 的首地址, 即数组中第一个元素 a[0] 的地址。

数组中 9 对应的是 a[8], 选项 B, P+8 表示数组中后 8 个元素的地址, 即 a[8] 的地址。*(P+8) 则表示该地址内所存放的内容, 即 a[8] 的值。

选项 A, *P 表示 P 所指向对象的内容, 此时 P 指向 a[0], *P 即 a[0] 的值 1。*P+9=1+9=10

选项 C, *P 表示 P 所指向对象的内容, 此时 P 指向 a[0], *P 即 a[0] 的值。因此 *P+=9 即 *P=*P+9, 等价于 a[0]=a[0]+9。

选项 D, P+8 表示数组中后 8 个元素的地址, 即 a[8] 的地址, 而非 a[8] 中的值。

39. 在 C 语言中, 以 D 作为字符串结束标志

A、'\n' B、' ' C、'0' D、'\0'

40. 下列数据中属于“字符串常量”的是 (A)。

A、“a” B、{ABC} C、'abc\0' D、'a'

41. 已知 char x[]="hello", y[]={ 'h', 'e', 'a', 'b', 'e' };, 则关于两个数组长度的正确描述是 B。

A、相同 B、x 大于 y C、x 小于 y D、以上答案都不对

C 语言中, 字符串后面需要一个结束标志位 '\0', 通常系统会自动添加。

对一维数组初始化时可采用字符串的形式 (例如本题数组 x), 也可采用字符集合的形式

(例如本题数组 y)。在以字符串形式初始化时，数组 x 不只要存储字符串中的字符，还要存储字符串后的结束标志位，因此数组 x 的长度为 6；在以字符集合形式初始化时，数组 y，仅存储集合中的元素，因此数组 y 长度为 5。

1. #include

```
main()
{ int a=1,b=3,c=5;
  if (c==a+b)
    printf("yes\n");
  else
    printf("no\n");
}
```

运行结果为: no

2. #include

```
main()
{ int a=12, b= -34, c=56, min=0;
  min=a;
  if(min>b)
    min=b;
  if(min>c)
    min=c;
  printf("min=%d", min);
}
```

运行结果为: min=-34

定义变量，并赋值 此时 a=12, b=-34, c=56, min=0

将 a 中值拷贝，赋给 min，覆盖了 min 中的 0，此时 min 中的值被更新为 12。

若 min>b 成立，则执行 min=b；若 min>c 成立，则执行 min=c；输出 min 中的值

12 大于 -34，第一个 if 语句的表达式成立，因此执行 min=b；执行后 min 中的值被更新为 -34。

-34 小于 56，第二个 if 语句的表达式不成立，因此不执行 min=c；

最后输出 min 中的值，为 -34。

3. #include

```
main()
{ int x=2,y=-1,z=5;
  if(x
  if(y<0)
    z=0;
  else
    z=z+1;
  printf( "%d\n", z);
}
```

运行结果为: 5

遇到选择结构，首先要明确条件表达式成立时执行哪些操作。本题中，第一个 if 语句，其后的复合语句没有大括号 {}，说明复合语句中只包含一条语句，进而省略了 {}。内层的 if...else... 是选择结构的第二种基本形式，在结构上视为一条语句。因此内层的

if...else... 作为第一个 if 语句的复合语句。

```
4. #include
main()
{ float a,b,c,t;
  a=3;
  b=7;
  c=1;
  if(a>b)
  {t=a;a=b;b=t;}
  if(a>c)
  {t=a;a=c;c=t;}
  if(b>c)
  {t=b;b=c;c=t;}
  printf("%.2f, %.2f, %.2f", a, b, c);
}
```

运行结果为: 1.00, 2.00, 7.00

本题包含了 3 个 if 语句, 每个 if 语句后的 {} 都不可省略, 因为每个 {} 中都包含了多条语句

若表达式 $a > b$ 成立, 则执行 {t=a;a=b;b=t;}

若表达式 $a > c$ 成立, 则执行 {t=a;a=c;c=t;}

若表达式 $b > c$ 成立, 则执行 {t=b;b=c;c=t;}

输出 a, b, c 中的值, 要求输出的每个数据宽度为 5 个空格, 小数部分保留 2 位, 数据右对齐

```
5. #include< stdio .h >
main ( )
{floatc=3.0 , d=4.0;
  if ( c>d ) c=5.0;
  else
  if ( c==d ) c=6.0;
  elsec=7.0;
  printf ( “%.1f\n”, c ) ;
}
```

运行结果为: 7.0

此题为 if...else... 语句的嵌套, 第二 if...else... 作为第一个 if...else... 语句 else 部分的复合语句。

若表达式 $c > d$ 成立, 则执行 $c=5.0$; 否则 (表达式 $c > d$ 不成立)

若表达式 $c == d$ 成立, 则执行 $c=6.0$; 否则, 执行 $c=7.0$; 输出 c 中的值

```
6. #include "stdio.h"
main()
{char c;
  c=getchar();
}
```



```
while(c!='?') {putchar(c);c=getchar();}
```

如果从键盘输入 abcde? fgh (回车)
运行结果为: abcde

7. #include

```
main()
{ char c;
while((c=getchar())!='$')
{ if( 'A' <=c&&c<= 'Z') putchar(c);
else if( 'a' <=c&&c<= 'z') putchar(c-32);}
}
```

当输入为 ab*AB%cd#CD\$时, 运行结果为: ABABCD

8. #include

```
main()
{ int x, y=0;
for(x=1;x<=10;x++)
{ if(y>=10)
break;
y=y+x;
}
printf(“%d %d”, y, x);
}
```

运行结果为: 10 5

```
for( 表达式1;表达式2;表达式3)
{
}
```

(1) 先求解表达式 1. (2) 求解表达式 2, 若其值为真, 执行循环体, 然后执行 (3). 若为假, 则结束循环, 转到 (5). (3) 求解表达式 3. (4) 转回上面 (2) 继续执行. (5) 循环结束, 执行 for 语句下面的一个语句 break, 跳出循环体; continue, 结束本次循环 (第 i 次循环), 继续执行下一次循环 (第 i+1 次循环)

9. #include

```
main( )
{char ch;
ch=getchar( );
switch(ch)
{case 'A' :printf(“%c”, 'A');
case 'B' :printf(“%c”, 'B'); break;
default: printf(“%s\n”, "other");
}}
```

当从键盘输入字母 A 时, 运行结果为: AB

10. #include

```
main( )
{ int a=1,b=0;
scanf(“%d”, &a);
```



```

switch(a)
{case 1: b=1; break;
case 2: b=2; break;
default : b=10; }
printf("%d ", b);
}

```

若键盘输入 5，运行结果为：10

首先用 scanf 函数为变量 a 赋值为 5。

执行 switch 语句。switch 后面的条件表达式为 a，因此表达式的值即为 5。用 5 依次与下面 case 中的常量匹配。没有找到匹配的常量，因此两个 case 后的语句都不执行。执行 default 后面的语句 b=10；将 10 赋给变量 b。输出变量 b，结果为 10

```

11. #include
main()_
{ char grade=' C' ;
switch(grade)
{
case 'A' : printf( "90-100\n" );
case 'B' : printf( "80-90\n" );
case 'C' : printf( "70-80\n" );
case 'D' : printf( "60-70\n" ); break;
case 'E' : printf( "<60\n" );
default : printf( "error!\n" );
}
}

```

运行结果为：

70-80

60-70

```

12. #include
main()
{ int y=9;
for(;y>0;y-- )
if(y%3==0)
{ printf("%d", --y);
}
}

```

运行结果为：

852

```

13. #include
main()
{ int i,sum=0;i=1;
do{ sum=sum+i; i++; }while(i<=10);
printf( "%d", sum);
}

```

运行结果为： 55

14. #include

```
#define N 4
main()
{ int i;
  int x1=1,x2=2;
  printf("\n");
  for(i=1;i<=N;i++)
  { printf("%4d%4d",x1,x2);
    if(i%2==0)
    printf("\n");
    x1=x1+x2;
    x2=x2+x1;
  }
}
```

运行结果为:

1 2 3 5

8 13 21 34

表达式 1 为 $i=1$, 表达式 2 (循环条件) 为 $i \leq N$ 即 $i \leq 4$, 表达式 3 为 $i++$
循环变量初值 i 为 1, 循环条件 (即表达式 2) $i \leq 4$ 成立, 进入第 1 次循环

第 1 次循环执行 $\text{printf}("%4d\%4d", x1, x2)$; 因此屏幕上输出 12

执行 if 语句。 $1\%2$ 不为 0, if 语句的条件表达式不成立, 不执行 $\text{printf}("\n")$;

执行 $x1=x1+x2=1+2=3$; 此时 $x1$ 中的值已变为 3 执行 $x2=x2+x1=2+3=5$ 。

转向表达式 3, 执行 $i++$, i 为 2。循环条件 $i \leq 4$ 成立, 进入第 2 次循环

第 2 次循环执行 $\text{printf}("%4d\%4d", x1, x2)$; 因此屏幕上输出 35

执行 if 语句。 $2\%2=0$, if 语句的条件表达式成立, 执行 $\text{printf}("\n")$; 执行

执行 $x1=x1+x2=3+5=8$; 此时 $x1$ 中的值已变为 8 执行 $x2=x2+x1=5+8=13$ 。

转向表达式 3, 执行 $i++$, i 为 3。循环条件 $i \leq 4$ 成立, 进入第 3 次循环

第 3 次循环执行 $\text{printf}("%4d\%4d", x1, x2)$; 因此屏幕上输出 813

执行 if 语句。 $3\%2$ 不为 0, if 语句的条件表达式不成立, 不执行 $\text{printf}("\n")$;

执行 $x1=x1+x2=8+13=21$; 此时 $x1$ 中的值已变为 21

执行 $x2=x2+x1=13+21=34$ 。

转向表达式 3, 执行 $i++$, i 为 4。循环条件 $i \leq 4$ 成立, 进入第 4 次循环

第 4 次循环

执行 $\text{printf}("%4d\%4d", x1, x2)$; 因此屏幕上输出 2134

执行 if 语句。 $4\%2=0$, if 语句的条件表达式成立, 执行 $\text{printf}("\n")$; 执行

执行 $x1=x1+x2=21+34=55$; 此时 $x1$ 中的值已变为 55

执行 $x2=x2+x1=34+55=89$ 。

转向表达式 3, 执行 $i++$, i 为 5。循环条件 $i \leq 4$ 不成立, 结束循环

15. #include

```
main()
{ int x, y;
  for(x=30, y=0; x>=10, y<10; x--, y++)
  x/=2, y+=2;
  printf("x=%d, y=%d\n", x, y);
}
```

```
}
```

运行结果为:

x=0, y=12

1. 输入 2 个整数，求两数的平方和并输出。

```
#include <stdio.h>
int main(void)
{
    int a, b, s;
    printf("please input a, b: \n");
    scanf("%d %d", &a, &b);
    s = a*a + b*b;
    printf("the result is %d \n", s);
    return 0;
}
```

2. 输入一个圆半径 r，当 r >= 0 时，计算并输出圆的面积和周长，否则，输出提示信息。

```
#include <stdio.h>
#define PI 3.14
int main(void)
{
    double r, area, girth;
    printf("please input r: \n");
    scanf("%lf", &r);
    if (r >= 0)
    {
        area = PI * r * r;
        girth = 2 * PI * r;
        printf("the area is %.2f \n", area);
        printf("the girth is %.2f \n", girth);
    }
    else
    {
        printf("Input error! \n");
        return 0;
    }
}
```

3、已知函数 $y=f(x)$ ，编程实现输入一个 x 值，输出 y 值。

$$y = \begin{cases} 2x+1 & (x < 0) \\ 0 & (x = 0) \\ 2x-1 & (x > 0) \end{cases}$$

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

```
void main()
```

```
{ int x,y;
```

```
scanf("%d",&x);
```

```
if(x<0) y=2*x+1;
```

```
else if(x>0) y=2*x-1;
```

```
else y=0;
```

```
printf("%d",y);
```

```
}
```

4.从键盘上输入一个百分制成绩 score，按下列原则输出其等级：score≥90，等级为 A；80≤score<90，等级为 B；70≤score<80，等级为 C；60≤score<70，等级为 D；score<60，等级为 E。

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

```
void main(){
```

```
int data;
```

```
char grade;
```

```
printf("Please enter the score:");
```

```
scanf("%d",&data);
```

```
switch(data/10)
```

```
{ case 10:
```

```
case 9: grade='A'; break;
```

```
case 8: grade='B'; break;
```

```
case 7: grade='C'; break;
```

```
case 6: grade='D'; break;
```

```
default: grade='E';
```

```
};
```

5. 编一程序每个月根据每月上网时间计算上网费用, 计算方法如下: ↵

$$\text{费用} = \begin{cases} 30\text{元} & \leq 10\text{小时} \\ \text{每小时3元} & 10 - 50\text{小时} \\ \text{每小时2.5元} & \geq 50\text{小时} \end{cases}$$

要求当输入每月上网小时数, 显示该月总的上网费用(6分) ↵

```
#include <stdio.h>↵  
void main()↵  
{ int hour;↵  
  float fee;↵  
  printf("please input hour:\n");↵  
  scanf("%d",&hour);↵  
  if(hour<=10)↵  
    fee=30;↵  
  else if(hour>=10&&hour<=50)↵  
    fee=3*hour;↵  
  else fee=hour*2.5;↵  
  printf("The total fee is %f",fee);↵  
}
```

6. 从键盘输入10个整数, 统计其中正数、负数和零的个数, 并在屏幕上输出。 ↵

```
#include <stdio.h>↵  
void main() {↵  
  int a, i,p=0,n=0,z=0;↵  
  printf("please input number");↵  
  for(i=0;i<10;i++){↵  
    scanf("%d",&a);↵  
    if(a>0) p++;↵  
    else if(a<0) n++;↵  
    else z++;↵  
  }↵  
  printf("正数: %5d, 负数: %5d, 零: %5d\n",p,n,z);↵  
}
```

7、编写程序实现求 1-10 之间的所有数的乘积并输出。

```
#include <stdio.h>
void main()
{
    int i;
    long sum=1;
    for(i=1; i<=10; i=i+1)
        sum=sum*i;
    printf("the sum of odd is :%d", sum);
}
```

8. 从键盘上输入 10 个数，求其平均值。

```
#include <stdio.h>
void main() {
    int a, i, sum=0;
    float ave;
    for(i=0; i<10; i++) {
        scanf("%d", &a);
        sum+=a;
    }
    ave=(float)sum/10;
    printf("ave = %f\n", ave);
}
```

9、编写程序实现求 1-1000 之间的所有奇数的和并输出。

```
#include <stdio.h>
void main() {
    int i, sum=0;
    for(i=1; i<1000; i=i+2)
        sum=sum+i;
    printf("the sum of odd is :%d", sum);
}
```

10. 有一个分数序列：2/1，3/2，5/3，8/5，13/8，.....编程求这个序列的前 20 项之和。

```
#include <stdio.h>
void main() {
    int i, t, n=20;
    float a=2, b=1, s=0;
    for(i=1; i<=n; i++) {
        s=s+a/b;
        t=a;
        a=a+b;
        b=t;
    }
    printf("sum=%6.2f", s);
}
```

第1题 m个人的成绩存放在 score 数组中，请编写函数 fun, 它的功能是：将低于平均分的人作为函数值返回，将低于平均分的分数放在 below 所指定的函数中。

```
int fun(int score[], int m, int below[])
{
    int i, k=0, aver=0;
    for(i=0; i<m; i++)
        aver+=score[i];
    aver/=m;
    for(i=0; i<m; i++)
        if(score[i]<aver)
        {
            below[k]=score[i];
            k++;
        }
    return k;
}
```

第2题 请编写函数 fun, 它的功能是：求出 1 到 100 之内能被 7 或者 11 整除，但不能同时被 7 和 11 整除的所有证书，并将他们放在 a 所指的数组中，通过 n 返回这些数的个数。

```
void fun(int *a, int *n)
{
    int i, j=0;
    for(i=2; i<1000; i++)
        if ((i%7==0 || i % 11==0) && i%77!=0)
            a[j++]=i;
    *n=j;
}
```

第3题 请编写函数 void fun(int x, int pp[], int *n), 它的功能是：求出能整除 x 且不是偶数的各整数，并按从小到大的顺序放在 pp 所指的数组中，这些除数的个数通过形参 n 返回。

```
void fun(int x, int pp[ ], int *n)
{
    int i=1, j=0, k=0, *t=pp;
    for(i=0; i<=x; i++)
        if(i%2!=0)
        {
            t[j]=i;
            j++;
        }
    for(i=0; i<j; i++)
        if(x%t[i]==0)
        {
            pp[k]=t[i];
            k++;
        }
}
```



```
*n=k;
```

```
}
```

第4题请编写一个函数 void fun(char *tt,int pp[]),统计在 tt 字符中"a"到"z"26 各字母各自出现的次数,并依次放在 pp 所指的数组中。

```
void fun(char *tt, int pp[])
```

```
{
```

```
int i;
```

```
for (i=0;i<26;i++)
```

```
pp[i]=0;
```

```
while (*tt)
```

```
{
```

```
switch (*tt)
```

```
{
```

```
case 'a' : pp[0]++;break;
```

```
case 'b' : pp[1]++;break;
```

```
case 'c' : pp[2]++;break;
```

```
case 'd' : pp[3]++;break;
```

```
case 'e' : pp[4]++;break;
```

```
case 'f' : pp[5]++;break;
```

```
case 'g' : pp[6]++;break;
```

```
case 'h' : pp[7]++;break;
```

```
case 'i' : pp[8]++;break;
```

```
case 'j' : pp[9]++;break;
```

```
case 'k' : pp[10]++;break;
```

```
case 'l' : pp[11]++;break;
```

```
case 'm' : pp[12]++;break;
```

```
case 'n' : pp[12]++;break;
```

```
case 'o' : pp[14]++;break;
```

```
case 'p' : pp[15]++;break;
```

```
case 'q' : pp[16]++;break;
```

```
case 'r' : pp[17]++;break;
```

```
case 's' : pp[18]++;break;
```

```
case 't' : pp[19]++;break;
```

```
case 'u' : pp[20]++;break;
```

```
case 'v' : pp[21]++;break;
```

```
case 'w' : pp[22]++;break;
```

```
case 'x' : pp[23]++;break;
```

```
case 'y' : pp[24]++;break;
```

```
case 'z' : pp[25]++;break;
```

```
}
```

```
tt++;
```

```
}
```

```
}
```

第5题请编写一个函数 void fun(int m,int k,int xx[]),该函数的功能是: 将大于整数 m

且紧靠 m 的 k 各素数存入 xx 所指的数组中。

```
void fun(int m, int k, int xx[])
{
    int g=0, l, j, flag=1;
    for(i=m+1; i<=m*m; i++)
    {
        for(j=0; j<l; j++)
        {
            if(i%j!=0)
                flag=1;
            else
            {
                flag=0;
                break;
            }
        }
        if (flag==1 && j>=i)
        {
            if (k>=0)
            {
                xx[g++]=i;
                k--;
            }
            else
                break;
        }
    }
}
```

第6题请编写一个函数 void fun(char a[], char b[], int n), 其功能是：删除以各字符串中指定下标的字符。其中，a 指向原字符串，删除后的字符串存放在 b 所指的数组中，n 中存放指定的下标。

```
void fun(char a[], char b[], int n)
{
    int l, j=0;
    for (i=0; i<LEN; i++)
    {
        if (i!=n)
        {
            b[j]=a[i];
            j++;
        }
        b[j]='\0' ;
    }
}
```

第7题请编写一个函数 int fun(int *s, int t, int *k), 用来求除数组的最大元素在数组中的下标并存放在 k 所指的储存单元中。

```

void fun (int *s, int t, int *k)
{
    int l, max;
    max=s[0];
    for(i=0;i<t;i++)
        if (s[i]>max)
        {
            max=s[i];
            *k=l;
        }
}

```

第8题编写函数 fun, 功能是：根据以下攻势计算 s, 计算结果作为函数值返回；n 通过形参传入。 $S=1+1/(1+2)+1/(1+2+3)+\dots+1/(1+2+3+4+\dots+n)$

```

float fun (int n)
{
    int i;
    float s=1.0, t=1.0;
    for(i=2;i<=n;i++)
    {
        t=t+i;
        s=s+1/t;
    }
    return s;
}

```

第9题编写一个函数 fun, 它的功能是：根据以下公式求 p 的值，结果由函数值带回。M 与 n 为两个正整数，且要求 $m>n$ 。 $p=m!/n!(m-n)!$

答案：

```

float fun(int m, int n)
{
    float p, t=1.0;
    int l;
    for (i=1;i<=m;i++)
        t=t*I;
    p=t;
    for(t=1.0, i=1;i<=n;i++)
        t=t*I;
    p=p/t;
    for(t=1.0, i=1;i<=m-n;i++)
        t=t*I;
    p=p/t;
    return p;
}

```

第10题编写函数 fun, 它的功能是：利用以下的简单迭代方法求方程 $\cos(x)-x=0$ 的一个实根。迭代步骤如下：

- (1)取 x_1 初值为 0.0; (2) $x_0=x_1$, 把 x_1 的值赋各 x_0 ;
(3) $x_1=\cos(x_0)$, 求出一个新的 x_1 ;
(4)若 x_0-x_1 的绝对值小于 0.000001, 则执行步骤(5), 否则执行步骤(2);
(5)所求 x_1 就是方程 $\cos(x)-x=0$ 的一个实根, 作为函数值返回。

程序将输出 root=0.739085。

```
float fun()  
{  
    float x1=0.0, x0;  
    do  
    {  
        x0=x1;  
        x1=cos(x0);  
    }  
    while(fabs(x0-x1)>1e-6);  
    return x1;  
}
```

- 1: 下列程序定义了 $n \times n$ 的二维数组, 并在主函数中自动赋值。请编写函数 fun(int a[][N]), 该函数的功能是: 使数组左下半三角元素中的值全部置成 0。

```
int fun(int a[][N])  
{  
    int I, j;  
    for(i=0; i<N; i++)  
        for(j=0; j<=i; j++)  
            a[i][j]=0;  
}
```

- 2: 下列程序定义了 $n \times n$ 的二维数组, 并在主函数中赋值。请编写函数 fun, 函数的功能使求出数组周边元素的平均值并作为函数值返回给主函数中的 s。

答案:

```
double fun(int w[][N])  
{  
    int I, j, k=0;  
    double s=0.0;  
    for(j=0; j<N; j++)  
    {  
        s+=w[0][j];  
        k++;  
    }  
    for(j=0; j<N; j++)  
    {  
        s+=w[N-1][j];  
        k++;  
    }  
    return s/k;  
}
```

```

k++;
}
for(i=1;i<=N-2;i++)
{
s+=w[i][0];
k++;
}
for(i=1;i<=N-2;i++)
{
s+=w[i][N-1];
k++;
}
return s/=k;
}

```

3: 请编写一个函数 void fun(int tt[m][n],int pp[n]),tt 指向一个 m 行 n 列的二维函数组, 求出二维函数组每列中最小元素, 并依次放入 pp 所指定一维数组中。二维数组中的数已在主函数中赋予。

```

void fun(int tt[M][N], int pp[N])
{
int I, j, min;
for(j=0;j<N;j++)
{
min=tt[0][j];
for(i=0;i<M;i++)
{
if (tt[i][j]<min)
min=tt[i][j];
}
pp[j]=min;
}
}

```

4: 请编写函数 fun, 函数的功能使求出二维数组周边元素之和, 作为函数值返回。二维数组中的值在主函数中赋予。

```

int fun(int a[M][N])
{
int I, j, s=0;
for(j=0;j<N;j++)
{
s+=a[0][j];
s+=a[M-1][j];
}
for(i=1;i<=M-2;i++)
{
s+=a[i][0];

```

```
s+=a[i][N-1];  
}  
return s;  
}
```

5: 请编写一个函数 unsigned fun(unsigned w), w 使一个大于 10 的无符号整数, 若 w 使 n ($n \geq 2$) 位的整数, 则函数求出 w 后 n-1 位的数作为函数值返回。

```
unsigned fun(unsigned w)  
{  
    unsigned t, s=0, sl=1, p=0;  
    t=w;  
    while(t>10)  
    {  
        if(t/10)  
            p=t;  
        s=s+p*sl;  
        sl=sl*10;  
        t=t/10;  
    }  
    return s;  
}
```

6: 请编写一个函数 float fun(double h), 函数的功能使对变量 h 中的值保留 2 位小树, 并对第三位进行四舍五入(规定 h 中的值位正数)。

```
float fun(float h)  
{  
    long t;  
    float s;  
    h=h*1000;  
    t=(h+5)/10;  
    s=(float)t/100.0;  
    return s;  
}
```

7: 请编写一个函数 fun(char *s), 该函数的功能使把字符串中的内容拟置。

```
void fun(char *s)  
{  
    char ch;  
    int l, m, n;  
    i=0;  
    m=n=strlen(s)-1;  
    while(i<(n+1)/2)  
    {  
        ch=s[i];  
        s[i]=s[m];  
        s[m]=ch;  
        i++;  
    }
```

```
m--;  
}
```

8: 编写程序, 实现矩阵(3 行 3 列)的转置(即行列互换)。

```
void fun(int array[3][3])
```

```
{  
    int l, j, temp;  
    for(i=0; i<3; i++)  
        for(j=0; j<l, j++)  
        {  
            temp=array[i][j];  
            array[i][j]=array[j][i];  
            array[j][i]=temp;  
        }  
}
```

9: 编写函数 fun, 该函数的功能是: 从字符串中删除指定的字符, 同一字母的大、小写按不同字符处理。

```
void fun(char s[], int c)
```

```
{  
    int i=0;  
    char *p;  
    p=s;  
    while(*p)  
    {  
        if(*p!=c)  
        {  
            s[i]=*p;  
            i++;  
        }  
        p++;  
    }  
    s[i]= '\0' ;  
}
```

10: 编写函数 int fun(int lim, int aa[max]), 该函数的功能是求出小于或等于 lim 的所有素数并放在 aa 数组中, 该函数返回所求的素数的个数。

```
int fun(int lim, int aa[MAX])
```

```
{  
    int k=0, l, j;  
    for(i=lim; i>1; i--)  
    {  
        for(j=2; j<i; j++)  
            if(i%j==0)  
                break;  
        else  
            continue;
```



```
if(j>=i)
{
aa[k]=i;
k++;
}
}
return k++;
}
```

1: 请编写函数 fun, 对长度位 7 个字符的字符串, 除首尾字符外, 将其余 5 个字符按 ascii 码降序排列。

```
void fun(char *s,int num)
{
char t;
int l,j;
for(i=1;i<num-2;i++)
for(j=i+1;j<num-1;j++)
if(s[i]<s[j])
{
t=s[i];
s[i]=s[j];
s[j]=t;
}
}
```

2: n 名学生的成绩已在主函数中放入一个带头节点的链表结构中, h 指向链表的头节点。请编写函数 fun, 它的功能是: 找出学生的最高分, 由函数值返回。

```
double fun(STREC *h)
{
double max;
STREC *q=h;
max=h->s;
do
{
if(q->s>max)
max=q->s;
q=q->next;
}
while(q!=0);
```

```
return max;
}
```

3: 请编写函数 fun, 该函数的功能是: 判断字符串是否为回文?若是则函数返回 1, 主函数中输出 yes, 否则返回 0, 主函数中输出 no。回文是指顺读和倒读都是一样的字符串。

```
int fun(char *str)
{
    int l, n=0; fg=1;
    char *p=str;
    while (*p)
    {
        n++;
        p++;
    }
    for(i=0; i<n/2; i++)
        if (str[i]!=str[n-1-i]);
        else
        {
            fg=0;
            break;
        }
    return fg;
}
```

4: 请编写一个函数 fun, 它的功能是: 将一个字符串转换为一个整数(不得调用 c 语言提供的将字符串转换为整数的函数)。

```
long fun (char *p)
{
    long s=0, t;
    int i=0, j, n=strlen(p), k, sl;
    if(p[0]=='-')
        i++;
    for(j=1; j<=n-1; j++)
    {
        t=p[j]- '0' ;
        sl=10;
        for(k=j; k<n-1; k++)
            t*=sl;
        s+=t;
    }
    if(p[0]=='-')
        return -s;
    else
        return s;
}
```

5 请编写一个函数 fun, 它的功能是: 比较两个字符串的长度, (不得调用 c 语言提供的求

字符串长度的函数)，函数返回较长的字符串。若两个字符串长度相同，则返回第一个字符串。

```
char *fun(char *s, char *t)
{
    char *p,*t1=t,*s1=s;
    int n=0;m=0;
    while (*s1)
    {
        n++;
        s1++;
    }
    while(*t1)
    {
        m++;
        t1++;
    }
    if(n>=m)
        p=s;
    else
        p=t;
    return p;
}
```

6: 请编写一个函数 fun，它的功能是：根据以下公式求 x 的值(要求满足精度 0.0005，即某项小于 0.0005 时停止迭代)：

$$x/2=1+1/3+1\times2/3\times5+1\times2\times3/3\times5\times7+1\times2\times3\times4/3\times5\times7\times9+\cdots+1\times2\times3\times\cdots\times n/3\times5\times7\times(2n+1)$$

程序运行后，如果输入精度 0.0005，则程序输出为 3.14...

```
double fun(double eps)
{
    double s;
    float n, t, pi;
    t=1;pi=0;n=1.0;s=1.0;
    while ((fabs(s))>=eps)
    {
        pi+=s;
        t=n/(2*n+1);
        s*=t;
        n++;
    }
    pi=pi*2;
    return pi;
}
```

7: 请编写一个函数 fun，它的功能是：求除 1 到 m 之内(含 m)能被 7 或 11 整除的所有整数放在数组 a 中，通过 n 返回这些数的个数。

```
void fun(int m, int *a, int *n)
```

```
{
    int I, j=0; *n=0;
    for(i=1; i<=m; i++)
        if (i%7==0 || i==0)
        {
            a[j]=I;
            j++;
        }
    *n=j;
}
```

8: 请编写一个函数 fun，它的功能是：找出一维整型数组元素中最大的值和它所在的下标，最大的值和它所在的下标通过形参传回。数组元素中的值已在主函数中赋予。主函数中 x 是数组名，n 是 x 中的数据个数，max 存放最大值，index 存放最大值所在元素的下标。

```
void fun (int a[], int n, int *max, int *d)
```

```
{
    int I;
    *max=a[0];
    *d=0;
    for(i=0; I<n; i++)
        if(a[i]>*max)
        {
            *max=a[i];
            *d=i;
        }
}
```

9: 请编写一个函数 fun，它的功能是：将 ss 所指字符串中所有下标为奇数位置上的字母转换为大写(若该位置上不是字母，则不转换)。

```
void fun(char *ss)
```

```
{
    int I, n;
    n=strlen(ss);
    for(i=1; i<n; i+=2)
        if(ss[i]>= 'a' && ss[i]<= 'z' )
            ss[i]=ss[i]-32;
}
```

10: 请编写一个函数 fun，它的功能是：求除一个 2×m 整型二维数组中最大元素的值，并将此值返回调用函数。

```
int fun(int a[][M])
```

```
{
    int I, j, max;
    max=a[0][0];
    for(i=0; i<2; i++)
```

```
for(j=0;j<M;j++)
if(a[i][j]>max)
max=a[i][j];
return max;
}
```

1: 请编写函数 fun, 其功能是: 将 s 所指字符串中除了下标为偶数、同时 ascii 值也为偶数的字符外, 其余的全都删除; 串中剩余字符所形成的一个新串放在 t 所指的一个数组中。

```
void fun(char *s, char t[])
{
int l, j=0, n;
n=strlen(s);
for(i=0; l, n; i++)
if(i%2==0&&s[i]%2==0)
{
t[j]=s[i];
j++;
}
t[j]= '\0' ;
}
```

2: 请编写函数 fun, 其功能是: 将 s 所指字符串中除了下标为奇数、同时 ascii 值也为奇数的字符之外, 其余的所有字符都删除, 串中剩余字符所形成的一个新串放在 t 所指的一个数组中。

```
void fun(char *s, char t[])
{
int l, j=0, n;
n=strlen(s);
for(i=0; l, n; i++)
if(i%2!=0&&s[i]%2!=0)
{
t[j]=s[i];
j++;
}
t[j]= '\0' ;
}
```

3: 假定输入的字符串中只包含字母和*号。请编写函数 fun, 它的功能是: 使字符串中尾部的*号不得多于 n 个; 若多于 n 个, 则删除多余的*号; 若少于或等于 n 个, 则什么也不做, 字符串中间和前面的*号不删除。

```
void fun(char *a, int n)
{
```

```

int i=0;k=0;
char *p, *t;
p=t=a;
while (*t)
t++;
t--;
while(*t== '*')
{
k++;
t--;
}
t++;
if(k>n)
{
while (*p&& p<t+n)
{
a[i]=*p;
i++;
p++;
}
a[i]= '\0' ;
}
}

```

4: 学生的记录由学号和成绩组成, n 名学生的数据已在主函数中放入结构体数组 s 中, 请编写函数 fun, 它的功能使: 把分数最高的学生数据放在 h 所指的数组中, 注意: 分数最高的学生可能不止一个, 函数返回分数最高的学生的人数。

```

int fun (STREC *a, STREC *b)
{
int I, j=0, n=0, max;
max=a[0].s;
for(i=0; i<N; i++)
if(a[i].s>max)
max=a[i].s;
for(i=0; i<N; i++)
if(a[i].s==max)
{
*(b+j)=a[i];
j++;
n++;
}
return n;
}

```

5: 请编写一个函数, 用来删除字符串中的所有空格。

```

void fun(char *str)

```

```

{
int i=0;
char *p=str;
while (*p)
{
if(*p!=' ' )
{
str[i]=*p;
i++;
}
p++;
}
str[i]='\0';
}

```

6: 假定输入的字符串中只包含字母和*号。请编写函数 fun，它的功能是：将字符串中的前导*号全部移到字符串的尾部。

```

void fun(char *a)
{
int i=0,n=0;
char *p;
p=a;
while (*p=='*')
{
n++;
p++;
}
while (*p)
{
a[i]=*p;
i++;
p++;
}
while(n!=0)
{
a[i]='*';
i++;
n--;
}
a[i]='\0';
}

```

7: 某学生的记录由学号、8 门课程成绩和平均分组成，学号和 8 门课程的成绩已在主函数中给出。请编写函数 fun，它的功能是：求出该学生的平均分放在记录的 ave 成员中。请自己定义正确的形参。

```

void fun(STREC *p)

```



```

{
double av=0.0;
int i;
for(i=0;i<N;i++)
av+=p->s[i];
av/=N;
p->ave=av;
}

```

8: 请编写函数 fun, 它的功能是: 求出 ss 所指字符串中指定字符的个数, 并返回此值。

```

int fun(char *ss, char c)
{
int n=0;
while (*ss)
{
if(*ss==c)
n++;
ss++;
}
return n;
}

```

9: 请编写函数 fun, 该函数的功能是: 移动一维数组中的内容, 若数组中由 n 个整数, 要求把下标从 0 到 p (p 小于等于 n-1) 的数组元素平移到数组的最后。

```

void fun(int *w, int p, int n)
{
int b[N], i, j=0;
for(i=0;i<=p;i++)
{
b[i]=w[i];
j++;
}
for(i=0;i<=p;i++)
{
w[j]=b[i];
j++;
}
}

```

10: 请编写函数 fun, 该函数的功能是移动字符串中内容, 移动的规则如下: 把第 1 到第 m 个字符, 平移到字符串的最后, 把第 m+1 到最后的字符移到字符串的前部。

```

void fun(char *w,int m)
{
char b[N];
int i,j=0;
for(i=0;i<m;i++)
{

```

```

b[j]=w[i];
j++;
}
for (i=0;i<strlen(w)-m;i++)
w[i]=w[i+m];
for(j=0;j<m;j++)
{
w[i]=b[j];
i++;
}
w[i]= '\0' ;
}

```

1: 第1题请编写函数 fun, 该函数的功能是: 将 m 行 n 列的二维数组中的字符数据, 按列的顺序依次放到一个字符串中。

```

void fun(char (*s)[n],char *b)
{int i,j,k=0;
for(j=0;j<n;j++)
for(i=0;i<m;i++)
{b[k]=*(s+i)+j)
k++;}
b[k]= '\0' ;}

```

2: 第2题下列程序定义了 $n \times n$ 的二维数组, 并在主函数中自动赋值。请编写函数 fun(int a[][n],int n), 该函数的功能是: 将数组右上半三角元素中的值乘以 m。

```

void fun(int a[][n],int m)
{int i,j;
for(j=0;j<n;j++)
for(i=0;i<=j;i++)
a[j]=a[j]*m;}

```

3: 第3题编写一个函数, 从传入的 num 个字符串中找出一个最长的一个字符串, 并通过形参指针 max 传回该串地址(用****作为结束输入的标志)。

```

char *fun(char (*a)[81],int num)
{int i;
char *max;
max=a[0];
for(i=0;i<num;i++)
if(strlen(max)<strlen(a))
max=a;
return max;}

```

4: 第4题编写一个函数, 该函数可以统计一个长度为 2 的字符串在另一个字符串中出现的次数。

```
int fun(char *str, char *substr)
```

```
{int n;  
char *p, *r;  
n=0;
```

```
while(*str)
```

```
{p=str;
```

```
r=substr;
```

```
while(*r)
```

```
if(*r==*p)
```

```
{r++;
```

```
p++;}
```

```
else
```

```
break;
```

```
if(*r== '\0' )
```

```
n++;
```

```
str++;}
```

```
return n;}
```

5: 第5题假定输入的字符串中只包含字母和*号。请编写函数 fun，它的功能是：只删除字符串前导和尾部的*号，串中字母之间的*号都不删除。形参 n 给出了字符串的长度，形参 h 给出了字符串中前导*号的个数，形参 e 给出了字符串中最后的*个数。在编写时不得使用 c 语言给提供得字符串函数。

```
void fun(char *a, int n, int h, int e)
```

```
{int i=0;
```

```
char *p;
```

```
for(p=a+h;p<a+n-e;p++)
```

```
{*(a+i)=*p;
```

```
i++;}
```

```
*(a+i)= '\0';}
```

6: 第6题学生得记录由学号和成绩组成，n 名大学生得数据已在主函数中放入结构体数组 s 中，请编写函数 fun，它的功能时：按分数的高低排列学生的记录，高分在前。

```
void fun(strec a[])
```

```
{int i, j;
```

```
strec t;
```

```
for(i=0; i<n-1; i++)
```

```
for(j=i+1; j<n; j++)
```

```
if(a[i].s<a[j].s)
```

```
{t=a;
```

```
a=a[j];
```

```
a[j]=t;}}
```

7: 第7题请编写一个函数 void fun(char *ss)，其功能时：将字符串 ss 中所有下标为奇数位置上的字母转换为大写(若位置上不是字母，则不转换)。

```
void fun(char *ss)
```

```
{int i, n=0;
```

```
char *p=ss;
```

```

while(*p)
{
    n++;
    p++;
}
for (i=0;i<n;i++)
    if ((ss= 'a' &ss<= 'z' )&i%2!=0)
        ss=ss-32;
ss= '\0' ;}

```

8: 第8题请编写函数 fun, 其功能是: 将两个两位数的正整数 a, b 合并成一个整数放在 c 中。合并的方式是: 将 a 数的十位和个位依次放在 c 数的千位和十位上, b 数的十位和个位数依次放在 c 数的百位和个位上。

```

void fun(int a,int b,long *c)
{
    *c=(a/10)*1000+(b/10)*100+(a)*10+b;
}

```

9: 第9题请编写函数 fun, 其功能是: 将 s 所指字符串中下标位偶数同时 ascii 值为奇数的字符删除, s 中剩余的字符形成的新串放在 t 所指的数组中。

```

void fun(char *s, char t[])
{
    int i, j=0, n=strlen(s);
    for(i=0;i<n;i++)
        if(i%2==0&s%2!=0)
            else
                {f[j]=s;
                j++;}
    t[j]= '\0' ;}

```

10: 第10题已知学生的记录是由学号和学习成绩构成, n 名学生的数据已存入 a 机构体数组中。请编写函数 fun, 该函数的功能是: 找出成绩最高的学生记录, 通过形参返回主函数(规定只有一个最高分)。

```

void fun(stu a[], stu *s)
{
    int i, max;
    max=a[0].s;
    for (i=0;i<n;i++)
        if(a. s>max)
            {max=a. s;
            *s=a;}
}

```

1: 第1题请编写函数 fun, 其功能是: 将所有大于 1 小于整数 m 的非素数存入 xx 所指的数组中, 非素数的个数通过 k 传回。

```

void fun(int m, int *k, int xx[])
{
    int i, j;
    int t=0;
    for(i=2;i<m;i++)
        {j=2;
        while(j<i)

```

```
{if(i%j==0)
```

```
{xx[t]=i;
```

```
t++;
```

```
break;}
```

```
j++;}
```

```
*k=t;}}
```

2: 第2题编写一个函数 fun, 它的功能是: 实现两个字符串的连接(不使用库函数 strcat), 即把 p2 所指的字符串连接到 p1 所指的字符串后。

```
void fun(char p1[],char p2[])
```

```
{int i=0,n=0;
```

```
char *p=p1,*q=p2;
```

```
while (*p)
```

```
{p++;
```

```
n++;}
```

```
i=n;
```

```
while(*p)
```

```
{p1=*q;
```

```
q++;
```

```
i++;}
```

```
p1= '\0' ;}
```

3: 第3题请编写函数 fun, 该函数的功能是: 实现 $b=a+a$, 即把矩阵 a 加上 a 的转置, 存放在矩阵 b 中。计算结果在 main 函数中输出。

```
void fun(int a[3][3],int b[3][3])
```

```
{int i,j,at[3][3];
```

```
for(i=0;i<=2;i++)
```

```
for(j=0;j<=2;j++)
```

```
at[j]=a[j];
```

```
for(i=0;i<3;i++)
```

```
for(j=0;j<3;j++)
```

```
b[j]=a[j]+at[j];}
```

4: 第4题学生的记录由学号和成绩组成, n 名学生的数据已在主函数中放入结构体数组 s 中, 请编写函数 fun, 它的功能是: 把低于平均分的学生数据放在 b 所指的数组中, 低于平均分的学生人数通过形参 n 传回, 平均分通过函数值返回。

```
double fun(strec *a, strec *b,int *n)
```

```
{double aver=0.0;
```

```
int i,j=0;
```

```
*n=0;
```

```
for (i=0;i<n;i++)
```

```
aver+=a.s;
```

```
aver/=n;
```

```
for(i=0;i<n;i++)
```

```
if(a.s<aver)
```

```
{b[j]=a;
```

```
(*n)++;
```

```
j++; }
```

```
return aver;}
```

5: 第5题请编写函数 fun, 该函数的功能是: 将 m 行 n 列的二维数组中的数据, 按行的顺序依次放到一维数组中, 一维数组中数据的个数存放在形参 n 所指的储存单元中。

```
void fun(int(*s)[10],int *b,int *n,int mm,int nn)
```

```
{int i,j;
```

```
for(i=0;i<mm;i++)
```

```
for(j=0;j<nn;j++)
```

```
{b[*n]=*(s+i+j);
```

```
*n=*n+1;}}
```

第6题假定输入的字符串中只包含字母和*号。请编写函数 fun, 它的功能是: 除了尾部的*号之外, 将字符串中其他*号全部删除。形参 p 已指向字符串中最后的一个字母。不使用 c 的字符串函数。

```
void fun(char*a,char *p)
```

```
{int i=0;
```

```
char *p=a;
```

```
while(q<=p)
```

```
{if(*q!='*')
```

```
{a=*q;
```

```
i++;}
```

```
q++;}
```

```
while(*q)
```

```
{a=*q;
```

```
i++;
```

```
q++;}
```

```
a='\0';}
```

第7题学生的记录是由学号和成绩组成, n 名学生的数据已在主函数中放入结构体数组 s 中, 请编写函数 fun, 它的功能是: 把指定分数范围内的学生数据放在 b 所指的数组中, 分数范围内的学生人数由函数值返回。

```
int fun(strec*a,strec *b,int l,int h)
```

```
{int i,j=0;
```

```
for(i=0;i<n;i++)
```

```
if(a.s>=l&&a.s<=h)
```

```
{b[j]=a;
```

```
j++;}
```

```
return j;}
```

第8题编写函数藏服那, 它的功能是: 求 n 以内(不包括 n)同时能被 3 与 7 整除的所有自然数之和的平方根 s, 并作为函数值返回。

```
double fun(int n)
```

```
{double s=0.0;
```

```
int i;
```

```
for(i=1;i<n;i++)
```

```
if(i%3==0&&i%7==0)
```

```
s=s+i;
```

```
s=sqrt(s);
```

```
return s;}
```

第9题请编写函数 fun，该函数的功能是：将放在字符串数组中的 m 个字符串（每串的长度不超过 n），按顺序合并组成一个新的字符串。

```
void fun(char a[m][n],char *b)
```

```
{int i,j,k=0;
```

```
for(i=0;i<m;i++)
```

```
{for(j=0;j<n;j++)
```

```
if(*(a+i)+j))
```

```
{b[k]=*(a+i)+j)
```

```
k++;}
```

```
else
```

```
break;
```

```
b[k]='\0';}}
```

第10题请编写函数 fun，该函数的功能是：删去一维数组中所有相同的数，使之只剩一个。数组中的数已按由小到大的顺序排列，函数返回删除后数组中数据的个数。

```
int fun(int a[],int n)
```

```
{int i,t,j=0,*p=a;
```

```
t=p[0];
```

```
for(i=0;i<=n;i++)
```

```
if(t==p)
```

```
;
```

```
else
```

```
{a[j]=t;
```

```
t=p;
```

```
j++;}
```

```
if(i>=n)
```

```
a[j]=t;
```

```
return j;}
```

第1题第请编写函数 fun，该函数的功能使：统计各年龄段的人数。N 个年龄通过调用随机函数获得，并放在主函数的 age 数组中；要求函数把 0 至 9 岁年龄段的人数放在 d[0]中，把 10 至 19 岁年龄段的人数放在 d[1]中，把 20 至 29 岁的人数放在 d[2]中，其余以此类推，把 100 岁(含 100 以上年龄的人数都放在 d[10]中。结果在主函数中输出。

```
void fun(int *a,int *b)
```

```
{int i,j;
```

```
for(j=0;j<m;j++)
```

```
b[j]=0;
```

```
for(i=0;i<n;i++)
```

```
if(a>=0&a<=9)
```



```

b[0]++;
else if(a>=10&a<=19)
b[1]++;
else if(a>=20&a<=29)
b[2]++;
else if(a>=30&a<=39)
b[3]++;
else if(a>=40&a<=49)
b[4]++;
else if(a>=50&a<=59)
b[5]++;
else if(a>=60&a<=69)
b[6]++;
else if(a>=70&a<=79)
b[7]++;
else if(a>=80&a<=89)
b[8]++;
else if(a>=90&a<=99)
b[9]++;
else
b[10]++;

```

第2题请编写函数 fun，该函数的功能是：统一一含字符串中单词的个数，作为函数值返回。一行字符串在主函数中输入，规定所有单词由小写字母组成，单词之间由若干个空格格开，一行的开始没有空格。

```

int fun (char *s)
{int i,n=0;
for(i=0;i<strlen(s);i++)
{if(s== 'a' &s<= 'z' &s[i+1]!=' '
|| s[i+1]== '\0' )
n++;}
return n;}

```

第3题请编写一个函数 fun，它的功能是：计算并输出给定整数 n 的所有因子(不包括 1 与自身)之和。规定 n 的值不大于 1000。

```

int fun(int n)
{int s=0,i;
for(i=2;i<=n-1;i++)
if(n%i==0)
s+=i;
return s;}

```

第4题请编写函数 fun，其功能是：将 s 所指字符串中 ascii 值为奇数的字符删除，串中剩余字符形成一个新串放在 t 所指的数组中。

```

void fun(char *s,char t[])
{int i,j=0,n;
n=strlen(s);

```

```
for(i=0;i<n;i++)
```

```
if(s%2==0)
```

```
{t[j]=s;
```

```
j++;}
```

```
t[j]='\0' ;}
```

5: 第5题请编写函数 fun, 其功能是: 将两位数的正整数 a、b 合并成一个整数放在 c 中。合并的方式是: 将 a 数的十位和个位数依次放在 c 数的百位和个位上, b 数的十位和个位数依次放在 c 数的十位和千位上。

```
void fun(int a,int b,long *c)
```

```
{*c=(b)*1000+(a/10)*100+(b/10)*10+a;}
```

1: 请编写函数 fun, 其功能时: 将两个两位数的正整数 a、b 合并形成一个整数放在 c 中, 合并的方式是: 将 a 数的十位和个位数依次放在 c 数的十位和千位上, b 的十位和个位数依次放在 c 数的百位和个位上。

```
void fun(int a,int b,long *c)
```

```
{*c=(a)*1000+(b/10)*100+(a/10)*10+b;}
```

2: 请编写函数 fun, 其功能是: 将 s 所指字符串中下标为偶数的字符删除, 串中剩余字符形成新串放在 t 所指数组中。

```
void fun(char *s, char t[])
```

```
{int i, j=0, n=strlen(s);
```

```
for(i=0;i<n;i++)
```

```
if(i%2!=0)
```

```
{t[j]=s;
```

```
j++;}
```

```
t[j]='\0' ;}
```

3: 假定输入的字符串中只包含字母和*号, 请编写函数 fun, 它的功能是: 除了字符串前导和尾部的*号之外, 将串中其他*号全部删除。形参 h 已指向字符串第一个字符, 形参 p 已指向字符串中最后一个字母。在编写程序时, 不得使用 c 语言提供的字符串函数。

```
void fun(char *a, char *h, char *p)
```

```
{int i=0;
```

```
char *q=a;
```

```
while(q<h)
```

```
{a=*q;
```

```
q++;
```

```
i++;}
```

```
while(q<p)
```

```
{if(*q!= '*')
```

```
{a=*q;
```

```
i++;}
```

```
q++;}
```

```
while(*q)
```

```
{a=*q;
i++;
q++;}
a= '\0' ;}
```

4: 学生的记录由学号和成绩组成 n 名学生的数据已在主函数中放入结构体数组 s 中, 请编写函数 fun, 它的功能时: 把分数最低的学生数据放在 h 所指的数组中, 注意: 分数最低的学生可能不止一个, 函数返回分数最低的学生的人数。

```
int fun(strec *a,strec *b)
{int i,j=0,n=0,min;
min=a[0].s;
for(i=0;i<n;i++)
if(a.s<min)
min=a.s;
for(i=0;i<n;i++)
if(a.s==min)
{*(b+j)=a;
j++;
n++;}
return n;}
```

5: 请编写函数 fun , 该函数的功能: 将 m 行 n 列的二维数组中的数据, 按列的顺序依次放到一维数组中。

```
void fun(int (*s)[]10,int *b,int *n,int mm,int nn)
{int i,j;
for(j=0;j<nn;j++)
for(i=0;i<mm;i++)
{b[*n]=*(s+i)+j);
*n=*n+1;}}
```

6: 请编写函数 fun, 其功能时: 计算并输出当 $x < 0.97$ 时下列多项式的值, 直到 $|s_n - s_{(n-1)}| < 0.000001$ 为止。

$$S_n = 1 + 0.5x + 0.5(0.5-1)/2!x(2) + \dots + 0.5(0.5-1)(0.5-2)\dots(0.5-n+1)/n!x(n)$$

```
double fun(double x)
{double s1=1.0,p=1.0,sum=0.0,s0,t=1.0;
int n=1;
do
{
s0=s1;
sum+=s0;
t*=n;
p*=(0.5-n+1)*x;
s1=p/t;
n++;}while(fabs(s1-s0)>1e-6);
return sum;}
```

7: 请编写一个函数 fun, 其功能是: 将两个两位数的正整数 ab 合并形成一个整数放在 c 中。合并方式是: 将 a 数的十位和个位数依次放在 c 数的个位和百位上, b 数的十位和个位数依次放在 c 数的十位和千位上。

```
void fun(int a,int b,long *c)
{*c=(b)*1000+(a)*100+(b/10)*10+a/10;}
```

8: 请编写函数 fun, 其功能是: 将 s 所指字符串中 ascii 值为偶数的字符删除, 串中剩余字符形成一个新串放在 t 所指的数组中。

```
void fun(char *s, char t[])
{int i, j=0, n=strlen(s);
for(i=0; i<n; i++)
if (s[i]%2!=0)
{t[j]=s[i];
j++;}
t[j]='\0' ;}
```

This document was truncated here because it was created using Aspose.Words in Evaluation Mode.