

一、选择题

1. 在每个 C 程序中都必须包含有这样一个函数，该函数的函数名为_____。
A. main B. MAIN C. name D. function
2. 在以下关于 C 语言的注释的叙述中，不正确的是_____。
A. 注释可用"/*"、"*/"形式表示，也可用"//"形式表示
B. 编译器在编译一个程序时，将跳过注释，不对其进行处理
C. 编译器在编译一个程序时，可发现注释中的单词拼写错误
D. 程序中注释的多少不会影响所生成的可执行文件的长度
3. 以下叙述不正确的是_____。
A. 在 C 程序中，严格区分大小写字母
B. 一个 C 源程序有且仅有一个 main 函数
C. 在 C 程序中，注释只能位于一条语句的后面
D. 一个 C 程序总是从 main 函数开始执行
4. 下列合法的标识符为_____。
A. abde+ B. #KDJF C. 67KDJ D. DK3_
5. 关于字符串和字符的关系正确的是_____。
A. "A" 与 'A' 是相同的
B. 字符串是常量，字符是变量
C. "A" 与 'A' 是不同的
D. "A" 与 "A" 是相同的
6. 下面关于语句的说法正确的是_____。
A. 下面这两段程序的执行效果是一致的

<pre>if (x>y) {z=x; s=z*z;} else {z=y; s=1/(z*z);}</pre>	<pre>if (x>y) z=x; s=z*z; else z=y; s=1/(z*z);</pre>
---	---

B. 空语句是只有一个分号的语句，它其实什么也不干
C. 语句的构成不一定需要分号
D. 语句的书写对程序的运行不起作用，因此可以随便写都可以
7. 以下各标识符中，合法的用户标识符组为_____。
A. PAd, P#d, b-b, abc, _0123, ssiped
B. cin, a10, CPP, float, del_word, signed
C. void, max, hiy, <fr>, list, *jer

- D. a, temp, _isw, INT, keep, wind
8. 字符串 “\”EOF= -\61\” 的长度是_____。
- A. 8 B. 9 C. 12 D. 非法字符串
9. 数学式 $\sqrt{|\sin x|}$ 在 C 程序中对应的不正确的表达式为_____。
- A. $\text{sqrt}(\sin(x)>0?\sin(x):-\sin(x))$ B. $\text{sqrt}(\text{fabs}(\sin x))$
C. $\text{sqrt}(\text{sqrt}(\sin x*\sin x))$ D. $\text{fabs}(\text{sqrt}(\sin(x)))$
10. 下列_____是 C 语言的有效标识符?
- A. _No1 B. No.1 C. 12345 D. int
11. C 语言中, 表示一条语句结束的标号是_____。
- A. # B. ; C. // D. }
12. 每个 C 程序中都必须有且仅有一个_____。
- A. 类 B. 预处理命令 C. 主函数 D. 语句
13. 下列数据类型不是 C 语言基本数据类型的是_____。
- A. 字符型 B. 整型 C. 实型 D. 数组
14. 下列字符列中, 可作为 C 语言程序自定义标识符是_____。
- A. switch B. file C. break D. do
15. 运算符 +、<=、=、% 中, 优先级最低的运算符是_____。
- A. + B. <= C. = D. %
16. 下列字符列中, 可以作为“字符串常量”的是_____。
- A. ABC B. "xyz" C. 'uvw' D. 'a'
17. 字符串"vm\x43\\\np\102q"的长度是_____。
- A. 8 B. 10 C. 17 D. 16
18. 在 C 语言中, 自定义的标识符_____。
- A. 能使用关键字并且不区分大小写
B. 不能使用关键字并且不区分大小写
C. 能使用关键字并且区分大小写
D. 不能使用关键字并且区分大小写
19. 设 x 和 y 均为逻辑型变量, 则 $x \&\& y$ 为真的条件是_____。
- A. 它们均为真 B. 其中一个为真
C. 它们均为假 D. 其中一个为假
20. 若给定条件表达式 $(m)?(a++):(a--)$, 则其中表达式 m_____。
- A. 和 $(m=0)$ 等价 B. 和 $(m=1)$ 等价
C. 和 $(m!=0)$ 等价 D. 和 $(m!=1)$ 等价
21. 在 if 语句中, 用作判断的表达式为_____。
- A. 关系表达式 B. 逻辑表达式

- C. 算术表达式 D. 任意表达式
22. 设 i 和 k 都是 int 类型, 则以下 for 循环语句_____。
- ```
for (i=0, k= -1; k=1; i++, k++) printf("****\n");
```
- A. 判断循环结束的条件不合法              B. 是无限循环  
C. 循环体一次也不执行                      D. 循环体只执行一次
23. 若有以下语句, 则输出结果为\_\_\_\_\_。
- ```
int a=2,b=1;  
string s1;  
s1= a>b?"a 大于 b":"a 小于 b";  
printf("%s\n", s1);
```
- A. a 大于 b B. a 小于 b C. 1 D. 输出语句有错
24. int k=10;
while(k=0) k=k-1;
则下面描述正确的是_____。
- A. while 循环执行 10 次 B. 循环是无限循环
C. 循环体语句一次也不执行 D. 循环体语句执行一次
25. int a=1,b=2,c=2,t;
while(a<b) {t=a;a=b;b=t;c--;}
printf("%d,%d,%d\n",a,b,c);
运行结果为_____。
- A. 1,2,0 B. 2,1,0 C. 1,2,1 D. 2,1,1
26. int x=-1;
do
{x=x*x;}
while(!x);
下列说法正确的是_____。
- 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
27. 以下能正确计算 $1*2*3*...*10$ 的程序段是_____。
- A. do {i=1;s=1;
 s=s*i; i++;}
 while(i<=10);
- B. do {i=1;s=0;
 s=s*i;i++;}
 while(i<=10);
- C. i=1;s=1;
 do {s=s*i;
 i++;}
 while(i<=10);
- D. i=1;s=0;
 do {s=s*i;
 i++;}
 while(i<=10);
28. 以下描述正确的是_____。
- A. 由于 do-while 循环中循环体语句只是一条可执行语句, 所以循环体内不能

使用复合语句

- B. do-while 循环由 do 开始，用 while 结束，在 while(表达式)后面不能写分号
- C. 在 do-while 循环体中，一定要有能使 while 后面表达式的值变为零（“假”）的操作

D. do-while 循环中，根据情况可以省略 while

29. while 和 do-while 循环的主要区别是_____。

- A. do-while 的循环体至少无条件执行一次
- B. while 的循环控制条件比 do-while 的循环控制条件严格
- C. do-while 允许从外部转到循环体内
- D. do-while 的循环体不能是复合语句

30. 下面有关 for 循环的正确描述是_____。

- A. for 循环只能用于循环次数已经确定的情况
- B. for 循环是先执行循环体语句，后判断表达式
- C. for 循环中，不能用 break 语句跳出循环体
- D. for 循环的循环体语句中，可以包含多条语句，但必须用花括号括起来

31. 对 for(表达式 1; ; 表达式 3)可理解为_____。

- A. for(表达式 1; 0; 表达式 3)
- B. for(表达式 1; 1; 表达式 3)
- C. for(表达式 1; 表达式 1; 表达式 3)
- D. or(表达式 1; 表达式 3; 表达式 3)

32. 在 C 编程语言中，以下正确的描述是_____。

- A. continue 语句的作用是结束整个循环的执行
- B. 只能在循环体内和 switch 语句体内使用 break 语句
- C. 在循环体内使用 break 语句或 continue 语句的作用相同
- D. 从多层循环嵌套中退出，只能使用 goto 语句

33. 下面程序段_____。

```
for(t=1;t<=100;t++)  
{scanf("%d",&x);  
  if(x<0) continue;  
  printf("%d\n",t);}
```

- A. 当 x<0 时整个循环结束
- B. x>=0 时什么也不输出
- C. cout 永远也不执行
- D. 最多允许输出 100 个非负整数

34. 定义一个函数实现交换 x 和 y 的值，并将结果正确返回。能够实现此功能的是_____。

- A. swap(int x, int y){ int t; t=x; x=y; y=t; }
- B. swap(int *x, int *y){ int t; t=x; x=y; y=t; }

- C. `swap(int *x, int *y){ int t; t= *x; *x = *y; *y=t; }`
D. `swap(int *x, int *y){ int *t; t=x; x=y; y=t; }`
35. 若有代数式 $3ae/bc$ ，则不正确的 C 语言表达式是_____。
- A. `a/b/c*e*3` B. `3*a*e/b/c`
C. `3*a*e/b*c` D. `a*e/c/b*3`
36. 以下表达式值为 3 的是_____。
- A. `16-13%10` B. `2+3/2`
C. `14/3-2` D. `(2+6)/(12-9)`
37. 设有说明语句，`int k=7, x=12;` 则以下能使值为 3 的表达式是_____。
- A. `x%=(k%=5)` B. `x%=(k-k%5)`
C. `x%= k-k%5` D. `(x%=k)-(k%=5)`
38. 若有条件表达式 `(exp)? A++: b--`，则以下表达式中能完全等价于表达式 `(exp)` 的是_____。
- A. `(exp==0)` B. `(exp!=0)` C. `(exp==1)` D. `(exp!=1)`
39. 设有以下语句：
- ```
typedef struct REC
{ char c; int a[4];} REC1;
```
- 则下面叙述中正确的是\_\_\_\_\_。
- A. 可以用 REC 定义结构体变量  
B. REC1 是 struct REC 类型的变量  
C. REC 是 struct 类型的变量  
D. 可以用 REC1 定义结构体变量
40. 下列关于结构的语法错误的是
- A. 结构是由用户自定义的一种数据类型  
B. 结构中可设定若干个不同数据类型的成员  
C. 结构中成员的数据类型可以是另一个已定义的结构  
D. 在定义结构时，可以为成员设置默认值
41. 以下关于结构体的叙述中，错误的是\_\_\_\_\_。
- A. 结构体是一种可由用户构造的数据类型  
B. 结构体中的成员可以具有不同的数据类型  
C. 结构体中的成员不可以与结构体变量同名  
D. 结构体中的成员可以是指向自身结构的指针类型
42. 说明一个结构体变量时，系统分配给它的内存是\_\_\_\_\_。
- A. 各成员所需内存量的总和  
B. 结构中第一个成员所需内存量  
C. 成员中占内存量最大者所需的容量  
D. 结构中最后一个成员所需内存量

43. 设有以下说明语句:

```
struct stu
```

```
{ int a; float b; } stutype;
```

则下面的叙述不正确的是\_\_\_\_\_。

- A. struct 是结构体类型的关键字
- B. struct stu 是用户定义的结构体类型名
- C. stutype 是用户定义的结构体类型名
- D. a 和 b 都是结构体成员名

44. 若有如下定义

```
struct data
```

```
{ int x,y; } test1={10,20}, test2;
```

则以下赋值语句中错误的是\_\_\_\_\_。

- A. test2=test1;
- B. test2.x=test1.x;
- C. test2.x=test1.y
- D. test2={30,40};

45. 以下对结构体变量 stu1 中成员 age 的非法引用是\_\_\_\_\_。

```
struct student
```

```
{ int age; int num; } stu1,*p;
```

```
p=&stu1;
```

- A. stu1.age
- B. student.age
- C. p->age
- D. (\*p).age

46. 当说明一个共用体变量时, 系统分配给它的内存是\_\_\_\_\_。

- A. 各成员所需内存量的总合
- B. 结构中第一个成员所需内存量
- C. 成员中占内存量最大者所需的容量
- D. 结构中最后一个成员所需内存量

47. 以下对 C 语言中共用体类型数据的叙述正确的是\_\_\_\_\_。

- A. 可以对共用体变量名直接赋值
- B. 一个共用体变量中可以同时存放其所有成员
- C. 一个共用体变量中不能同时存放其所有成员
- D. 共用体类型定义中不能出现结构体类型的成员

48. 若有以下定义和语句:

```
union data
```

```
{ int i; char c; float f; } a;
```

```
int n;
```

则以下语句正确的是\_\_\_\_\_。

- A. a=5;
- B. a={2,'a',1.2}
- C. printf("%d\n",a);
- D. n=1;

49. 设有以下说明, 则下面不正确的叙述是\_\_\_\_\_。

union

```
{ int i; char c; float f; } un;
```

- A. un 所占的内存长度等于成员 f 的长度
- B. un 的地址和它的成员地址都是同一地址
- C. un 可以作为函数参数
- D. 不能对 un 赋值, 但可以在定义 un 时对它初始化

50. C 语言共用体类型变量在程序运行期间

- A. 所有成员一直驻留在内存中
- B. 只有一个成员驻留在内存中
- C. 部分成员驻留在内存中
- D. 没有成员驻留在内存中

51. 使用 typedef 定义一个新类型的正确步骤是\_\_\_\_\_。

- A. (2)、(4)、(1)、(3)      B. (1)、(3)、(2)、(4)
- C. (2)、(1)、(4)、(3)      D. (4)、(2)、(3)、(1)

- (1) 把变量名换成新类型名
- (2) 按定义变量的方法写出定义体
- (3) 用新类型名定义变量
- (4) 在最前面加上关键则 typedef

52. 下面对 typedef 的叙述中不正确的是\_\_\_\_\_。

- A. 用 typedef 可以定义各种类型名, 但不能用来定义变量
- B. 用 typedef 可以增加新类型
- C. 用 typedef 只是将已存在的类型用一个新的标识符来代表
- D. 使用 typedef 有利于程序的通用和移植

53. 下面试图为 double 说明一个新类型名 real 的语句中, 正确是\_\_\_\_\_。

- A. typedef real double;
- B. typedef double real;
- C. typedef real=double;
- D. typedef double=real;

54. 下面能正确进行字符串赋值操作的是\_\_\_\_\_。

- A. char s[5]={“ABCDE”};      B. char s[5]={‘A’, ‘B’, ‘C’, ‘D’, ‘E’};
- C. char \*s; s=“ABCDE”;      D. char \*s; scanf(“%s”,s);

55. 下面程序段的运行结果是\_\_\_\_\_。

```
char *s="abcde";
```

```
s+=2; printf(“%d”,s);
```

- A. cde      B. 字符 c      C. 字符 c 的地址      D. 无确定的输出结果

56. 下面程序段的运行结果是\_\_\_\_\_。

```
char str[]="ABC", *p=str;
```

```
printf("%d\n", *(p+3));
```

A. 67    B. 0    C. 字符 C 的地址    D. 字符 C

57. 下面程序段的运行结果是\_\_\_\_\_。

```
char a[]="language", *p;
```

```
p=a;
```

```
while(*p!='u'){printf("%c",*p-32); p++;}
```

A. LANGUAGE    B. language    C. LANG    D. langUAGE

58. 有如下说明语句, 则以下不正确的叙述是\_\_\_\_\_。

```
char a[]="It is mine";
```

```
char *p="It is mine";
```

A. a+1 表示的是字符 t 所在存储单元的地址

B. p 指向另外的字符串时, 字符串的长度不受限制

C. p 变量中存放的地址值可以改变

D. a 中只能存放 10 个字符

答案:

1. A    2. C    3. C    4. D    5. C    6. B    7. D    8. A  
9. B    10. A    11. B    12. C    13. D    14. B    15. C    16. B  
17. A    18. D    19. A    20. C    21. D    22. B    23. A  
24. C    25. D    26. C    27. C    28. C    29. A    30. D    31. B  
32. B    33. D    34. C  
35. C    36. B    37. D    38. B    39. D    40. D    41. C    42. A  
43. C    44. D    45. B    46. C    47. C    48. B    49. C    50. B    51. C  
52. B    53. B    54. C    55. C    56. B    57. C    58. D

## 二、填空题

- 程序的三种基本的控制结构是: \_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_。
- C 语言中的每条基本语句以\_\_\_\_\_作为结束符, 每条复合语句以\_\_\_\_\_作为结束符。
- 关系表达式  $x+y>5$  的相反表达式为\_\_\_\_\_。
- 下面的程序段中循环体的执行次数是\_\_\_\_\_。  

```
a=10;b=0;
do{b+=2;a-=2+b;} while(a>=0);
```
- 下面程序段的运行结果为\_\_\_\_\_。  

```
x=2;
```



- do{printf("\*"); x--; } while(!x==0);
6. 已知 `int a[3][2]={3, 2, 1};` 则表达式 “`a[0][0]/a[0][1]`” 的值是\_\_\_\_\_。
  7. 若用数组名作为函数调用时的实参, 则实际上传递给形参的是\_\_\_\_\_。
  8. 如果一个函数无返回值, 定义它时函数类型应是\_\_\_\_\_。
  9. 预处理命令以\_\_\_\_\_开头。
  10. 指针类型变量用于存储\_\_\_\_\_, 在内存中它占有一个存储空间。
  11. 赋值语句在执行时先计算\_\_\_\_\_, 再将该值转换成与\_\_\_\_\_一致的值, 并将值赋给左部变量。
  12. 在程序中 `else` 总是与\_\_\_\_\_的 `if` 配对, 而不管程序的书写格式如何。
  13. 在函数中使用\_\_\_\_\_语句返回值。
  14. 函数的形参在未被调用前\_\_\_\_\_分配空间, 函数形参的\_\_\_\_\_要和实参相同。
  15. 函数调用一般分\_\_\_\_\_, \_\_\_\_\_, 前者的特点是形参和实参各有独立的存储空间, 形参的存储空间在函数被调用时才分配, 形参内容的变化不影响实参, 后者的特点是形参实参共用一个存储空间, 形参内容的改变可以改变实参值。
  16. 在传值调用中, 实际是把实参的\_\_\_\_\_传给形参。
  17. `int a[3]=3;` 则 `a[2]=_____`。
  18. `int a[3]={1, 3, 5, 7};` 则\_\_\_\_\_。
  19. 数组由\_\_\_\_\_和\_\_\_\_\_构成, 其数组元素的数据类型是\_\_\_\_\_。
  20. 声明长度为 10 的 `float` 型数组 `a`, 且初始化为 0 的语句是: \_\_\_\_\_。
  21. 指针是存放一个变量的\_\_\_\_\_的变量。
  22. 当数组名作形参时, 相当于\_\_\_\_\_。
  23. 要访问数组元素可以采用\_\_\_\_\_, \_\_\_\_\_、\_\_\_\_\_。
  24. 定义一个指针 `p`, 它指向一个有 6 个整型元素的一维数组, 定义语句为\_\_\_\_\_。
  25. 定义一个指针数组 `p`, 它有 4 个元素, 每个指针指向一个整型数据, 定义语句为\_\_\_\_\_。
  26. \_\_\_\_\_。
  27. 指针可以指向函数, 这时它所指的是函数的\_\_\_\_\_。
  28. C 语言的源程序必须通过\_\_\_\_\_和\_\_\_\_\_后, 才能被计算机执行。
  29. C 语言源程序文件的后缀是\_\_\_\_\_; 经过编译后, 生成文件的后缀是\_\_\_\_\_; 经过连接后, 生成文件的后缀是\_\_\_\_\_。
  30. 在 C 语言中, 表示逻辑 “真” 值用\_\_\_\_\_。

31. 设 y 为 int 型变量, 请写出描述 “y 是奇数” 的表达式\_\_\_\_\_。
32. C 语言提供的 3 种逻辑运算符是\_\_\_\_、\_\_\_\_和\_\_\_\_\_。
33. 若 x、y、z 均为 int 型变量, 则描述 “x 或 y 中有一个小于 z” 的表达式是\_\_\_\_\_。
34. 设 a、b、c 均为 int 型变量且 a=7.5, b=2, c=3.6, 则表达式 a>b && c>a || a<b && !c>b 的值是\_\_\_\_\_。
35. 设 a、b、c 均为 int 型变量且 a=6, b=4, c=2, 则表达式!(a-b)+c-1 && b+c/2 的值是\_\_\_\_\_。
36. 设 a、b 均为 int 型变量且 a=2, b=4, 则表达式!(x=a) || (y=b) && 0 的值是\_\_\_\_\_。
37. 设 a、b、c 均为 int 型变量且 a=1, b=4, c=3, 则表达式!(a<b) || !c && 1 的值是\_\_\_\_\_。
38. 若有条件 “ $2 < x < 3$  或  $x < -10$ ”, 其对应的 C 语言表达式是\_\_\_\_\_。

答案:

1. 顺序结构 选择结构 循环结构
2. ; }
3.  $x+y \leq 5$
4. 3
5. \*\*
6. 1
7. 数组首地址
8. 内置函数
9. #
10. 所指变量的地址
11. 表达式的值 左部变量类型
12. 它前面最近且未配对
13. return
14. 不 类型及数量
15. 传值调用 传址调用
16. 值
17. 0
18. 会出错
19. 数组名 下标 相同的
20. float a[10]={0};

21. 地址
22. 指针
23. 下标法    地址法    指针法
24. `int (*p)[6]`
25. `int *p[4]`
26. 首地址
27. 入口地址
28. 编译    连接
29. `c`   `obj`    `exe`
30. 非 0 数字
31. `(y%2) == 1`
32. `&&    ||    !`
33. `x < z    ||    y < z`
34. 0
35. 1
36. 0
37. 0
38. `x < 3    &&    x > 2    ||    x < -10`

### 三、填空完成下面的程序

1. 以下程序的功能是：输出 x, y, z 三个数中的最大者。请填空。

```
#include <stdio.h>
int main()
{ int x=4, y=6, z=7;
 int u,v;
 if(x>y) u=x;
 else u=y;
 if(u>z) v=u;
 else v=z;
 printf("v=%d",v);
 return 0;
}
```

2. 以下程序的功能是：输入 3 个整数，按从大到小的顺序进行输出。请填空。

```
#include <stdio.h>
int main()
{ int x,y,z,c;
 scanf("%d%d%d",&x,&y,&z);
```

```
if(__y<z__)
 { c=y;y=z;z=c;}
if(__x<z)
 { c=x;x=z;z=c;}
if(__x<y)
 { c=x;x=y;y=c;}
printf(“%d,%d,%d”,x,y,z);
return 0;
}
```

3. 以下程序的功能是：输入一个字符，如果它是一个大写字母，则把它变成小写字母；如果它是一个小写字母，则把它变成大写字母；其他字符不变。请填空。

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

```
int main()
```

```
{ char ch;
```

```
scanf(“%c”,&ch);
```

```
if(__ch>='A' && ch<='Z'__) ch=ch+32;
```

```
else if(__ch>='a' && ch<='z'__) __ch=ch-32;__
```

```
printf(“%c”,ch);
```

```
return 0;
```

```
}
```

4. 以下程序的功能是：根据输入的三角形的三条边，判断是否能组成三角形，若可以则输出它的面积和三角形的类型。请填空。

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

```
#include<math.h>
```

```
int main()
```

```
{ float a,b,c;
```

```
float s,area;
```

```
scanf(“%f%f%f”,&a,&b,&c);
```

```
if(__a+b>c && b+c>a && a+c>b__)
```

```
{ s=(a+b+c)/2;
```

```
area=sqrt(s*(s-a)*(s-b)*(s-c));
```

```
printf(“%f”,area);
```

```
if(__a==b && b==c__)
```

```
printf(“等边三角形”);
```

```
else if(__ (a==b && b!=c) || (a==c && c!=b) || (c==b && a!=c) __)
```

```
printf(“等腰三角形”);
```

```
else if((a*a+b*b=c*c) || (a*a+c*c=b*b) || (b*b+c*c=a*a))
```

```
 printf("直角三角形");
 else printf("一般三角形");
 }
 else printf("不能组成三角形");
 return 0;
}
```

5. 将以下含有 switch 语句的程序段改写成对应的含有非嵌套 if 语句的程序段。  
请填空。

含有 switch 语句的程序段:

```
int x,y,m;
y=(int)(x/10);
switch(y)
{ case 10: m=5;break;
 case 9: m=4;break;
 case 8: m=3;break;
 case 7: m=2;break;
 case 6: m=1;break;
 default: m=0;
}
```

含有非嵌套 if 语句的程序段:

```
int x,m;
if(x<110 && x>=100) m=5;
if(x<100 && x>=90) m=4;
if(x<90 && x>=80) m=3;
if(x<80 && x>=70) m=2;
if(x<70 && x>=60) m=1;
if(x<60 || x>=110) m=0;
```

6. 根据以下给出的嵌套 if 语句, 填写对应的 switch 语句, 使它完成相同的功能。

(假设 mark 的取值在 1—100 之间。)

if 语句:

```
if(mar<60) k=1;
else if(mark<70) k=2;
else if(mark<80) k=3;
else if(mark<90) k=4;
else if(mark<=100) k=5;
```

switch 语句:

```
switch ((int)(mark/10))
```

```
{case 0: case 1: case 2: case 3: case 4: case 5: k=1;break;
case 6: k=2; break;
case 7: k=3;break;
case 8: k=4;break;
case 9: case 10: k=5;
}
```

7. 下面程序的功能是从键盘输入若干学号，然后输出学号中百位数字是 3 的学号（输入 0 时结束循环），请填空。

```
#include <stdio.h>
int main()
{ long int num;
 scanf("%ld",&num);
 do
 { if (num/100%/10==3) printf("%ld",num);
 scanf("%ld",&num);
 } while (num!=0);
 return 0;
}
```

8. 下面程序的功能是计算正整数 2345 的各位数字的平方和，请填空。

```
#include <stdio.h>
int main()
{ int n=2345, sum=0;
 do
 { sum=sum+ (n%10)*(n%10);
 n= n/10 ;
 }while(n);
 printf("sum=%d",sum);
 return 0;
}
```

9. 下面程序的功能是求算式  $xyz + yzz = 532$  中的  $x, y, z$  的值（其中  $xyz$  和  $yzz$  分别表示一个三位数），请填空。

```
#include <stdio.h>
int main()
{ int x,y,z,I,result=532;
 for(x=1;x<10;x++)
 for(y=1;y<10;y++)
 for(z=0 ; z<10; z++)
```

```
 { i=100*x+10*y+z+100*y+10*z+z;
 if (i==result)
 printf("x=%d,y=%d,z=%d\n",x,y,z);
 }
 return 0;
}
```

10. 有 1020 个西瓜，第一天卖一半多两个，以后每天卖剩下的一半多两个，下面的程序统计买完所需的天数，请填空。

```
#include <stdio.h>
int main()
{ int day=0, x1=1020, x2;
 while (x1)
 { x2= x1/2-2 ;
 x1=x2;
 day++;
 }
 printf("day=%d\n",day);
 return 0;
}
```

11. 下面的程序的功能是用 do-while 语句求 1—1000 之间满足“用 3 除余 2，用 5 除余 3，用 7 除余 2”的数，且一行只打印 5 个数，请填空。

```
#include <stdio.h>
int main()
{ int i=1,j=0;
 do
 { if (i%3==2 && i%5==3 && i%7==2)
 { printf("%4d",i);
 j++;
 if (j%5==0) printf("\n");
 }
 i=i+1;
 } while(i<1000);
 return 0;
}
```

12. 下面程序的功能是求 1000 以内的所有完全数。请填空。（说明：一个数如果恰好等于它的因子之和（除自身外），则称该数为完全数。例如：6=1+2+3，6 为

完全数)

```
#include <stdio.h>
int main()
{ int a,k,m;
 for(a=1;a<=1000;++a)
 { for(m=0,k=1;k<=a/2;k++) if(!(a%k)) m=m+k;
 if(m==a) printf("%4d",a);
 }
 return 0;
}
```

13. 下面的程序是从键盘输入的 10 个整数中,找出一个能被 7 正处的数,若找到,打印次数;若未找到,打印 not exist。请填空。

```
#include <stdio.h>
int main()
{ int k,a;
 for(k=1;k<=10;k++)
 { scanf("%d",&a);
 if(a%7==0) break;
 }
 if(k<=10) printf("%d\n",a);
 else printf("not exist\n");
 return 0;
}
```

#### 四、给出下列程序运行后的输出结果

1. 当 a=1,b=3,c=5,d=4 时,执行以下程序段后 x 的值是 2。

```
if (a<b)
 if(c<d) x=1;
else
 if(a<c)
 if(b<d) x=2;
 else x=3;
 else x=6;
```



```
else x=7;
```

2. 以下程序的输出是\_\_\_\_\_ -1 \_\_\_\_\_。

```
#include <stdio.h>
int main()
{ int a=100,x=10,y=20,ok1=5,ok2=0;
 if(x<y)
 if(y!=10)
 if(!ok1)
 a=1;
 else
 if(ok2) a=10;
 a=-1;
 printf("%d\n",a);
 return 0;
}
```

3. 以下程序的输出是\_\_\_\_\_ 2 \_\_\_\_\_。

```
#include <stdio.h>
int main()
{ int x=2,y=-1,z=2;
 if(x<y)
 if(y<0) z=0;
 else z+=1;
 printf("%d\n",z);
 return 0;
}
```

4. 以下程序的输出是\_\_\_\_\_ OK \_\_\_\_\_。

```
#include <stdio.h>
int main()
{ int x=1;
 if(x=2)
 printf("OK");
 else if(x<2) printf("%d\n",x);
 else printf("Quit");
 return 0;
}
```

5. 若运行时为变量 x 输入 12，则以下程序的运行结果是\_\_\_\_\_ 0 \_\_\_\_\_。

```
#include <stdio.h>
int main()
{ int x, y;
 scanf("%d",&x);
 y=x>12? X+10: x-12;
 printf("%d\n",y);
 return 0;
}
```

6. 若运行时输入: 2<回车>, 则以下程序的运行结果是 2nd class postage is 14p。

```
#include <stdio.h>
int main()
{ char class;
 printf("Enter 1 for 1st class post or 2 for 2nd post");
 scanf("%c",&class);
 if(class == '1')
 printf("1st class postage is 19p");
 else
 printf("2nd class postage is 14p");
 return 0;
}
```

7. 若运行时输入 4.4<回车>, 则以下程序的运行结果是 Selling Price(0.3)\$ 5.72。

```
#include <stdio.h>
int main()
{ float costPrice, sellingPrice;
 printf("Enter costPrice $: ");
 scanf("%f",&costPrice);
 if(costPrice>=5)
 { sellingPrice=costPrice+costPrice*0.25;
 printf("Selling Price(0.25)$ %6.2f",sellingPrice);
 }
 else
 { sellingPrice=costPrice+costPrice*0.3;
 printf("Selling Price(0.3)$ %6.2f",sellingPrice);
 }
 return 0;
}
```

8. 若运行时输入: 1605<回车>, 则以下程序的运行结果是 4: 05PM。

```
#include <stdio.h>
int main()
{ int t,h,m;
 scanf("%d",&t);
 h=(t/100)%12;
 if(h==0) h=12;
 printf("%d: ",h);
 m=t%100;
 if(m<10) printf("0");
 printf("%d",m);
 if(t<1200 || t==2400) printf("AM");
 else printf("PM");
 return 0;
}
```

9. 以下程序的运行结果是 20。

```
#include <stdio.h>
int main()
{ int a,b,c,d,x;
 a=0;
 c=0;
 b=1;
 d=20;
 if(a) d=d-10;
 else if(!b)
 if(!c) x=15;
 else x=25;
 printf("%d\n",d);
 return 0;
}
```

10. 以下程序的输出结果是\_\_\_\_\_。

```
#include <stdio.h>
int main()
{ int x,y;
 x=5;
 switch(x)
 { case 1:
```

```
 case 2:
 case 3:
 case 4: printf("x<5\n");
 case 5: printf("x=5\n");
 default: printf("The value of x is unknown.\n");
 return 0;
}
```

输出结果:

x=5

The value of x is unknown.

11. 下面程序的运行结果是 123。

```
#include <stdio.h>
int main()
{ int num=0;
 while(num<=2)
 { num++;
 printf("%d",num);
 }
 return 0;
}
```

12. 若运行以下程序时，从键盘输入 3.6 2.4<回车>，则下面程序的运行结果是 1.600000。

```
#include <stdio.h>
#include <math.h>
int main()
{ float x,y,z;
 scanf("%f%f",&x,&y);
 z=x/y;
 while(1)
 { if(fabs(z)>1.0 {x=y; y=z; z=x/y;}
 else break;
 printf("%f\n",y);
 }
 return 0;
}
```

13. 下面程序的运行结果是 8。

```
#include <stdio.h>
int main()
```

```
{ int a,b;
 for(a=1, b=1; a<=100; a++)
 { if(b>=20) break;
 if(b%3==1) {b+=3;continue;}
 b -=5;
 }
 printf("%d\n",a);
 return 0;
}
```