

电大资源网 11253 《C 语言程序设计》国家开放大学期末考试题库 (854) [笔试+一平台机考]

适用:【笔试+一平台机考】【试卷号: 11253】【课程号: 02274】

总题量 (854): 单选(259) 判断(223) 填空(129) 编程题(39) 程序填空题(178) 函数功能(26)

作者: 电大资源网: http://www.dda123.cn/ (任何问题可微信留言, 搜微信: 905080280)

资料考前整理, 只供大家复习使用! 题库上次考试可用, 这次有可能改版, 如果科目改版资料对不上, 可以把科目名称发我微信, 可退回下载该改版科目的积分

ps: 如果把改版科目可用的题目拍图发微信可奖励 10-20 积分, 把最题库发微信可奖励 20-50 积分

单选(259)--电大资源网: http://www.dda123.cn/ (微信搜: 905080280)

- 1、char 类型的长度为 ( )。-->[A.1](#)
- 2、char 类型的长度为 ( ) 个字节。-->[A.1](#)
- 3、C 程序中的哪些是预处理指令: ( )。-->[A. #includestdio. h](#)
- 4、C 语言 if 语句中的表达式是: ( )。-->[A. 可以是任意表达式](#)
- 5、C 语言程序由函数组成, 它的 ( )。-->[B. 主函数可以在其它函数之后, 函数内不可以嵌套定义函数。](#)
- 6、C 语言程序中的基本功能模块为 ( )。-->[A.函数](#)
- 7、C 语言源程序的基本单位是 ( )。-->[B. 函数](#)
- 8、C 语言源程序文件的缺省扩展名为 ( )。-->[D. .c](#)
- 9、C 语言中 while 和 do-while 循环的主要区别是: ( )。-->[D. do-while 的循环体至少执行一次](#)
- 10、C 语言中打开一个数据文件的系统函数为 ( )。-->[D. fopen \( \)。](#)
- 11、C 语言中的系统函数 fopen ( ) 是 ( ) 一个数据文件的函数。-->[D.打开](#)
- 12、C 语言中函数返回值的类型是由 ( ) 决定的。-->[A. 函数定义时指定的类型](#)
- 13、C 语言中普通整型变量 int 在内存中占 ( ) 字节。-->[B. 2](#)
- 14、C 语言中使用的字符常量, 其起止标记符是 ( )。-->[A.单引号](#)
- 15、C 语言中用于结构化程序设计的 3 种基本结构是 ( ) -->[A. 顺序结构、选择结构、循环结构](#)
- 16、C 语言属于 ( ) 语言。-->[C. 高级语言](#)
- 17、float 类型的长度为 ( ) C.4
- 18、for 循环语句“for (i=0; i<n; i+=2) S;”中循环体 S 语句被执行的次数为 ( )。-->[A. \(n+1\) /2](#)
- 19、for 循环语句能够被改写为 ( )。-->[D.while 语句](#)
- 20、for 循环语句能够被改写为 ( ) 语句。-->[D.while](#)
- 21、int 类型的长度为 ( )。C.4
- 22、x>0&&x<=10 的相反表达式为 ( )。-->[A. x=0x10](#)
- 23、x>5 的相反表达式为 ( )。-->[D.x=5](#)
- 24、被调函数调用结束后, 返回到: ( )。-->[D. 主调函数中该被调函数调用语句的下一语句](#)
- 25、标准输出设备显示器的文件流标识符是 ( )。B.stdout
- 26、标准输入设备的文件流标识符是 ( )。-->[B.stdin](#)
- 27、标准输入设备键盘的文件流标识符是 ( )。-->[A. stdin](#)
- 28、表示文件结束符的符号常量 EOF 的值为 ( )。D.-1
- 29、表示文件结束符的符号常量为 ( )。-->[C.EOF](#)

30、不符合 C 语言规定的复合语句是 ( )。-->[D. {y=10}](#)

31、程序的运行结果是: ( )。

```
#include<stdio.h>
inttest (intb)
{b*=10;
printf (“b=%d”,b) ;
returnb/2;
voidmain ( )
{intb=60;
b=test (b) ;
printf (“b=%d”,b) ;
}
```

程序的运行结果是: ( [D.b=600b=300](#) )。

- 32、程序运行中需要从键盘上输入多于一个数据时, 各数据之间默认使用 ( ) 符号作为分隔符。-->[D.空格或回车](#)
- 33、程序运行中需要从键盘上输入多了一个数据时, 若不特别规定分隔符, 则输入的各数据之间所使用的分隔符为 ( )。D. 空格或回车
- 34、程序运行中需要从键盘上输入多于一个数据时, 各数据之间默认使用 ( ) 符号作为分隔符。-->[D. 空格或回车](#)
- 35、程序运行中需要从键盘上输入多于一个数据时, 各数据之间应使用的分隔符为 ( )。-->[D. 空格或回车](#)
- 36、从一个数据文件中读人以换行符结束的一行字符串的函数为 ( )。-->[B.fgets \( \)。](#)
- 37、带有随机函数调用的表达式 rand ( ) %20 的值在 ( ) 区间内。-->[C. 0~19](#)
- 38、当处理特定问题时的循环次数已知时, 通常采用 ( ) 循环来解决。-->[A. for](#)
- 39、当处理特定问题时的循环次数已知时, 通常采用的语句是 ( )。-->[A. for](#)
- 40、短整数类型的长度为 ( ) B.2
- 41、对二维数组的正确定义是: ( )。-->[C. inta\[ \]\[3\]={1.2.3.4.5.6};](#)
- 42、对于一个长度为 n 的字符串, 保存它至少需要占用的存储字节数为 ( )。-->[C.n+1](#)
- 43、对于一个二维字符数组 a[M][N], 存储每个字符串的长度至多为 ( )。D.N-1
- 44、假定 a 为一个数组名, 则下面表达式中错误的是 ( )。B. \*a++
- 45、假定 a 为一个数组名, 在下面的表达式中, 存在语法错误的是 ( )。-->[B. \\*a++](#)
- 46、假定 a 为一个数组名, 则下面存在错误的表达式为的 ( )。-->[B. \\*a++](#)
- 47、假定 a 为一个整数类型的数组名, 整数类型的长度为 4, 则元素 a[4] 的地址比 a 数组的首地址大 ( ) 个字节。-->[C.16](#)
- 48、假定 a 为一个字符数组名, 则元素 a[i] 的指针访问方式为 ( )。-->[B.\\* \(a+i\)。](#)
- 49、假定 i 的初值为 0, 则在循环语句“while (i<n) (s+=i\*i; i++;)”中, 其循环体被执行的次数为 ( )。-->[B.n](#)
- 50、假定 k 是一个 double 类型的变量, 则定义变量 p 的正确语句为 ( )。-->[D. char\\*p=“Thankyou!”;](#)
- 51、假定 n 的值为 5, 则表达式 n++ 的值为 ( )。-->[B.5](#)

52、假定 n 的值为 6, 则表达式 ++n 的值为 ( )。-->[D.7](#)

53、假定 n 的值为 6 则表达式 n++ 的值为 ( )。-->[A.6](#)

54、假定 p 是一个指向 float 类型的数据指针, 则 p+1 所指向的数据的地址比 p 所指向的数据的地址大 ( ) 个字节。-->[B.2](#)

55、假定 p 是一个指向 float 型数据的指针, 则 p+1 所指数据的地址比 p 所指数据的地址大 ( ) 个字节。-->[C. 4](#)

56、假定 p 是一个指向 float 型数据的指针, 则 p+1 所指数据的地址比 p 所指数据的地址增加的字节数为 ( )。-->[C. 4](#)

57、假定 p 是一个指向数据对象的指针, 则通过 p 表示该数据对象为 ( )

D.\*p  
58、假定 p 所指对象的值为 25, p+1 所指对象的值为 42, 贝 iJ 表达式 \*p++ 的值为 ( )。A.25

59、假定 s 被定义为指针类型 char\* 的变量, 初始指向的字符串为“Helloworld!”, 若要使变量 P 指向 s 所指向的字符串, 则 P 应定义为 ( )。-->[A.cha\\*P=S;](#)

60、假定 x 的值为 5,y 的值为 6, 则表达式 x+\*-y 的值为 ( )。-->[A.25](#)

61、假定变量 m 的定义为“int m=7;”, 则下面语句格式正确的是 ( )。

B.\*a++  
62、假定变量 m 定义为“intm=7;”, 则下面正确的语句为 ( )。

-->[B. int\\*p=&m;](#)

63、假定变量定义为“intm=7;”, 则下面定义语句格式正确的是 ( )。B. int \* p= &m;

64、假定的总次数为 ( 的初值为。), 则在循环语句“while (i<n) (s+=i\*i; i++;)”中循环体被执行-->[B.n](#)

65、假定对二维数组 a[3][1] 进行初始化的数据为

{{3,5,6},{2,8},{7}}, a[1][1] 被初始化为 ( )。D.8

66、假定二维数组的定义语句为“int \*a[3][5];”, 则该数组所占用的存储空间的字节数为 ( )。D.60

67、假定是一个指向 float 型数据的指针, 则 p+1 所指数据的地址比所指向数据的地址大 ( )。C. 个字节

68、假定要访问一个结构指针变量 x 所指对象中的数据成员 a, 则表示方法为 ( ) B. x->a

69、假定要访问一个结构变量 x 中的数据成员 a, 则表示方法为 ( ) .A.x.a

70、假定要访问一个结构变量中的由指针成员所指向的对象, 则表示方法为 ( )。D.\* (x.a)

71、假定要访问一个结构指针变量 x 中的数据成员 a, 则表示方法为 ( ) B.x->a

72、假定要访问一个结构指针变量 x 中的数据成员 a, 则表示方法为 ( )。-->[B.x.a](#)

73、假定一堆数组的定义为“char.a[8];”, 则该数组所占用的存储空间的字节数为 ( )。D. 32

74、假定一个二维数组的定义为“inta[3][4]=《3,4》, (2,8,6)”;。则元素 a[1][0] 的值为 ( )。-->[B.2](#)

75、假定一个二维数组的定义语句为“inta[3][4]={3,4},{2,8,6}”;。则元素 a[1][1] 的值为 ( )。-->[D. 8](#)

76、假定一个二维数组的定义语句为“inta[3][4]={3,4},{2,8,6}”;。则元素 a[1][2] 的值为 ( )。-->[C. 6](#)

77、假定一个二维数组的定义语句为“`inta[3][4]={{3,4},{2,8,6}};`”, 则元素 `a[2][1]` 的值为 ( )。-->A. 0

78、假定一个函数的二维数组参数说明为 `charw[][N]`, 与之等价的指针参数说明为 ( )。-->A. `char (*w)[N]`

79、假定一个函数的数组参数说明为 `chara[]`, 与之等价的指针参数说明为 ( )。-->char\*a

80、假定一个函数的原型语句为“`intff (int*x);`”, 一个整型数组为 `a[10]`, 则下面函数调用表达式不正确的是 ( )。-->B. `ff(a[0])`。

81、假定一个函数的原型语句为“`int ff (int.x);`”, 一个整型数组为 `a[10]`, 则下面函数调用表达式不正确的是 ( )。B. `ff(a[0])`

82、假定一个函数定义为“`staticintf1 (intx,inty) {returnx+y;}`”, 该函数名称为 ( )。-->C. `f1`

83、假定一个函数原型为“`char*func`”, 则该函数的返回类型为 ( )。B. `int*`-->D. `char*`

84、假定一个函数原型语句为“`int f1 (inta[], int n);`”, 与数组参数等价的表示为 ( ) D. `int*a`

85、假定一个结构类型的定义为“`structA{inta,b; doublec;};`”, 则该类型的长度为 ( )。-->D. 16

86、假定一个结构类型的定义为“`structD{inta; D*next;};`”, 则该类型的长度为 ( )。-->B. 8

87、假定一个链表的表头指针为 `f`, 结点中包含有 `data` 和 `next` 域, 则向该链表的表头插入一个地址为 `p` 的结点时, 应执行的操作为 ( )。-->B. `p-next=f` 和 `f-p`

88、假定一个链表中结点的结构类型为“`structAA{intdata,structAA*next;};`”, 则 `next` 数据成员的类型为 ( )。-->B. `structAA*`

89、假定一维数组的定义为“`char*a[8];`”, 则该数组所占用的存储空间的字节数为 ( )。-->D.32

90、假定有“`structBOOK{chartitle[40]; floatprice; }book;`”, 则正确的语句为 ( )。-->B. `structBOOK*x=&book;`

91、假定有“`structBOOK{chartitle[40]; floatprice; }; structBOOKbook;`”, 则不正确的语句为 ( )。-->A. `structBOOK*x=malloc (book)`。

92、假定有定义为“`inta[10],x,*pa=a;`”, 若要把数组 `a` 中下标为 3 的元素值赋给 `x`, 则不正确的赋值为 ( )。-->D. `x=*pa+3`

93、假定有定义为“`intm=7,*p;`”, 则给 `p` 赋值的正确表达式为 ( )。-->B. `p=&m`

94、假定有一个定义语句为“`int a[10]= (3, 4, 2, 8, 6);`”, 则元素 `a[3]` 的值为 ( ) C.8

95、假定有一个定义语句为“`inta[10]= (3,4,2,8,6);`”, 则元素 `a[5]` 的值为 ( )。-->A.0

96、假定有一个元素类型为 `int` 的数组 `a[10]`, 它所占用的存储空间的字节数为 ( )。C.40

97、假定有一条定义语句为“`intm=7,*p;`”, 则给 `p` 赋值的正确表达式为 ( )。-->B. `p=&m`

98、假定有语句“`intb[10], *pb;`”, 则不正确的赋值为 ( )。-->B. `pb=b[5]`

99、假定有语句为“`int *p=malloc(10+20,sizeof(int));`”, 则 `P` 所指向的动态数组中所包含的元素个数为 ( )。

答: C.30

100、假定有语句为“`intb[10]; int*pb;`”, 则下面不正确的赋值语句为 ( )。-->D. `pb=b[5];`

101、假定有语句为“`int, p=malloc (20,sizeof (int));`”, 则 `p` 所指向的动态数组中所包含的元素个数为 ( )。-->C.20

102、假定有语句为“`int*p=malloc (10+20, sizeof (int));`”, 则 `p` 所指向的动态数组中所包含的元素个数为 ( )。-->D. 40

103、假定有语句为“`inta[4][51]= ( (1,3,5), (2,8));`”, 则 `a[1][1]` 的值为 ( )。-->C.8

104、假定整数类型的指针 `P` 所指数数据单元的值为 30, `p-h1` 所指数数据单元的值为 40, 则执行“`*p++`”, `p` 所指数数据单元的值为 ( )。-->A.40

105、假定指针变量 `p` 定义为“`int*p=malloc (sizeof (int));`”, 要释放 `p` 所指向的动态存储空间应调用的函数为 ( ) A. `free (p)`

106、将两个字符串连接起来组成一个字符串时, 选用的字符串函数为 ( )。-->C. `strcat ()`。

107、逻辑表达式“`(x>0&&x<=10)`”的相反表达式为 ( )。A. `x<=0 || x>10`

108、枚举类型中的每个枚举常量的值都是一个 ( )。-->A. 整数

109、每个 C 程序文件在编译时可能出现有警告性错误, 其对应的标识符号为 ( )。-->A. `warning`

110、每个 C 语言程序文件的编译错误分为 ( ) 类。-->B.2

111、能把函数处理结果的两个数据返回给主调函数, 在下面的方法中不正确的是 ( )。-->A. `return` 这两个数

112、判定逻辑值为“真”的最准确叙述是 ( )。-->C. 非 0 的整数

113、请写出上面程序的运行结果: ( )。  
#include<stdio.h>  
#defineN6  
voidmain ()  
inti,a[N]={2,5,8,10,15,21};  
for (i=0;i<N;i++)  
if (a[i]%5) printf ("%d",a[i]);  
printf ("\n");  
请写出上面程序的运行结果: (B.2821)。

114、如果 `inta=3,b=4;` 则条件表达式“`a<b? a: b`”的值是: ( )。-->A. 3

115、如果一个函数作为表达式被调用, 则该函数必须是 ( )。-->A. 有返回值的函数

116、若“`intn; floatf=13. 8;`”, 则执行“`n= (int) f%3`”后, `n` 的值是 ( )。-->A. 1

117、若 `n` 的值为 10, 则执行“`for (i=1; i<n; i++) S;`”语句的过程中, 表达式 `i=1` 共被执行的次数为 ( )。A.1

118、若 `P` 指向 `x`, 则与的等价表示为 ( )。B. 个字节

119、若需要把一个字符串赋给字符数组, 则选用的函数是 ( )。-->D. `strcpy`

120、若要使 `p` 指向二维整型数组 `a[10][20]`, 则 `p` 的类型为 ( )。-->D. `int (*) [20]`

121、若要以读和写两种操作方式打开一个二进制文件, 当文件不存在时返回打开失败信息, 则选用的打开方式字符串为 ( )。-->C. “`rb+`”

122、若以下选项中的变量已正确定义, 则正确的赋值语句是 ( ) -->C. `x3=0*12;`

123、若用数组名作为函数调用的实参, 传递给形参的是 ( )。-->A. 数组的首地址

124、若有说明语句“`inta[5],*p=a;`”, 则对数组元素的正确引用是: ( )。-->C. `*(p+2)`

125、若有一个函数原型为“`double*function ()`”, 则其返回值类型为 ( )。-->B. 实数指针型

126、若有一个函数原型语句为“`dCble-funCion ()`”, 则此函数的返回值类型为 ( )。-->C. 函数指针型

127、若有以下数组说明, 则 `i=10; a[a[i]]` 元素数值是 ( )。-->C. 6

128、若有语句为“`inta[10],x,*pa=a;`”, 要把数组 `a` 中下标为 3 的元素值赋给 `x`, 则不正确的语句为 ( )。-->D. `x=*pa+3;`

129、上面程序的运行结果是: ( )。#include<stdio.h>  
#include<stdio.h>  
voidmain ()  
{chardata[5];  
inti;  
for (i=0;i<5;i++)  
data[i]='A'+i;  
for (i=0;i<5;i=i+2)  
data[i]=data[i]+32;  
for (i=0;i<5;i++)  
printf ("%d", data[i]);  
}  
上面程序的运行结果是: (D.aBcDe)。

130、设 `x` 和 `y` 均为逻辑值, 则 `x&&.y` 为真的条件是 ( )。-->A. 它们均为真

131、设 `x` 和 `y` 均为逻辑值, 则 `xy` 为假的条件是: ( )。-->C. 它们均为假

132、设 `x` 和 `Y` 均为逻辑值, 则 `x 衣 aY` 为真的条件是 ( )。-->A. 它们均为真

133、设有两条语句为“`ina=3; a+=a*a;`”, 则执行结束后, `a` 的值为 ( )。-->A.12

134、设有两条语句为“`inta=12; a+=a*a;`”, 则执行结束后, `a` 的值为 ( )。-->C. 156

135、设有语句“`inta=12-->156`

136、设有语句“`inta=12; a+=a*a;`”, 则执行结束后, `a` 的值为 ( )。-->C. 156

137、十进制数 50 表示成符合 C 语言规定的八进制数为 ( )。-->D. 062

138、为了避免嵌套的 `if-else` 语句的二义性, C 语言规定 `else` 总是与 ( ) 组成配对关系。-->C. 在其之前最近的未配对的 `if`

139、无论在何种操作系统及 C 程序开发环境下, 运行 C 程序都必须按顺序经过 ( ) 三个步骤之后程序才能运行。-->B. 编辑编译连接

140、下列表达式中, 与下标引用 `a[k]` 等效的是: ( )。-->A. `*(a+k)`

141、下列各语句定义了数组，其中哪一个是不正确的 ( )。

--> C. float x[2][1]={1,2,4,6,8,10};

142、下列关于 `break` 语句的描述中，不正确的是 ( )。--> C. `break` 语句可用于 `if` 体内，它将退出 `if` 语句

143、下列关于 C 语言的说法错误的是 ( )。--> B. C 语言不区分大小写。

144、下列结论中，只有 ( ) 是不正确的。--> D. C 语言所有函数都可以相互调用，包括调用 `main`。

145、下列数值中最大值是 ( ) C. 28%8

146、下列四组选项中，正确的 C 语言标识符是 ( )。--> C. `a123`

147、下列一组初始化语句中，正确的是 ( ) --> B. `inta[8]={0,2,4,0,1}`

148、下列指针的定义不正确的定义是 ( )。--> A. `int *p=&i;i;`

149、下面 ( ) 表达式的值为 4。--> D. `(int) (11.0/3+0.5)`

150、下面的保留字不能作为函数返回类型的是 ( )。 C. `sizeof`

151、下面的函数原型声明中存在语法错误的是 ( )。

C. `AA(int a; int b;)`

152、下面的描述错误的是 ( )。--> C. 函数内可以嵌套定义函数

153、下面循环语句执行结束后输出的 `i` 值为 ( )。

`for(int i=0;i<n;i++)if(i>n/2){cout<<i<<endl; break;}`

答案: B. `n/2+1`

154、下面运算符优先级最高的是 ( )。 B. []

155、向一个二进制文件中写入信息的函数为 ( )。--> D. `fwrite` ( )。

156、向一个二进制文件中写入信息的函数 `fwrite` ( ) 带有的参数个数为 ( )。--> D. 4

157、向一个二进制文件中写入信息的函数为 ( )。--> D. `fwrite` ( )

158、向一个文本文件中写入一个字符的系统函数为 ( )。 A. `fputc` ( )

159、向一个文本文件中写入一个字符的系统函数为 ( )。 A. `fgetc` ( )

160、循环体至少被执行一次的循环语句为 ( )。--> C. `do...while`

161、循环语句“`for (i=0; i<=n; i++) S;`”中循环体 `S` 被执行的次数为 ( )。--> D. `n+1`

162、要利用 `const` 保留字定义整型常量 `ABC` 的值为 10，则对应的定义语句为 ( )。--> D. `const int ABC=10;`

163、要使字符串变量 `str` 具有初值“`happy`”，正确的初始化语句有 ( )。--> C. `char str[]="happy";`

164、一个 C 语言程序必须包含一个 ( ) 函数。--> B. `main`

165、一个 C 语言的源程序中 ( ) --> A. 必须有一个主函数

166、一个程序文件开始使用的每条预处理命令，其首字符必须是 ( ) B. `#`

167、一个程序中用户定义的普通函数的名称不能为 ( )。 B. `main`

168、一个二维字符数组 `a[M][N]` 能够存储的字符串个数至多为 ( ) A. `M`

169、一个函数定义所包含的两个部分是 ( )。 A. 函数头和函数体

170、一个指针类型的简单变量占用内存中的字节数为 ( ) B. 4

171、已知 `inta[3][4]`；则对数组元素引用正确的是 ( )。--> C. `a[2][0]`

172、已知 `x` 的初值为 15，则下列各表达式中的最大值为 ( )。

--> B. `++x`

173、已知 `x` 的初值为 15，则下列各表达式中的最小值为 ( )。

--> C. `x/=2`

174、已知：`inta[10]`；则对 `a` 数组元素的正确引用是 ( )。--> D. `a[0]`

175、已知“`char a[3][5]`；”，则元素 `a[2][3]` 的地址比该数组的首地址大 ( )。 A. 13 个字节

176、已知“`int a[10], x, *pa=a;`”，若要把数组 `a[3]` 元素的值赋给 `x`，则不正确的语句为 ( )。 D. `x=*pa+3;`

177、已知“`int b[10]; int *pb;`”，则下面不正确的赋值语句为 ( ) D. `pb=b[5]`

178、已知“`inta[10], x, *pa-a;`”，若要把数组 `a[5]` 元素的值赋给 `x`，则不正确的语句为 ( )。--> D. `x=*pa+5;`

179、已知“`int *p=malloc (100);`”，要释放 `p` 所指向的动态内存，正确的语句为 ( )。--> A. `free (p)`。

180、已知“`char a[3][5]`；”，则元素 `a[2][3]` 的地址比该数组的首地址大 ( ) A. 13 个字节

181、已知数组 `a` 的定义为 `char a[]="abc\tden"`；，请写出 `strlen (a)` 的值为 ( )。--> A. 7

182、以下不能定义为用户标识符的是 ( ) --> D. `scanf`

183、以下程序的输出是 ( ) `int main () {`

```
int main()
{
    int i, sum=0;
    for(i=0; i<=100; i++) sum=sum+i;
    printf("sum=%d\n", sum);
    return 0;
}
```

输出: C. 5050

184、以下程序的输出为 ( ) `#include`

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

```
main()
```

```
{int a=1,b=3,c=5;
```

```
if(c==a+b)
```

```
printf("yes\n");
```

```
else
```

```
printf("no\n");
```

```
输出: B. no
```

185、以下程序的输出为 ( ) `#include`

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

```
void main()
```

```
{int x=2,y=-1,z=5;
```

```
if(x<y)
```

```
if(y<0)
```

```
z=0;
```

```
else
```

```
z=z+1;
```

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

```
输出: A. 5
```

186、以下对一维数组 `a` 的正确说明是 ( )。

--> D. `char a[3]={'a','b','c'};`

187、以下关于循环的描述中，错误的是 ( )。--> D. `do...while`

语句与 `while` 语句的区别仅仅是关键字 `while` 的位置不同

188、以下关于循环体的描述中，错误的是 ( )。--> C. 循环体中不能出现 `goto` 语句

189、以下哪个不属于 C 语言的特点: ( )。--> D. 编写的程序不可移植

190、以下哪个是 C 语言的输出语句: ( )。--> C. `printf`

191、以下哪个是算法表示的方法: ( )。--> D. 以上均是

192、用 `calloc` 函数创建具有 10 个整型元素的动态存储空间的正确调用是 ( )

C. `calloc(10, sizeof(int))`

193、用 `calloc` 函数创建具有 10 个整型元素的一维数组的正确语句是 ( )。--> C. `int *p=calloc (10,4)`。

194、用 `Callo` 函数创建具有 10 个整型元素的动态存储空间的正确调用是 ( )。--> B. `calloc (10)`。

195、用于从键盘上为变量输入值的标准输入函数是 ( ) C. `scanf` ( )

196、用于输出表达式值的标准输出函数是 ( )。--> C. `printf` ( )

197、由 C 语言源程序文件编译而成的目标文件的扩展名为 ( )。

C. `obj`

198、由 C 语言目标文件链接而成的可执行文件的默认扩展名为 ( )。

--> B. `exe`

199、由 C 语言源程序文件编译而成的目标文件的缺省扩展名为:

( )。--> D. `obj`

200、有如下程序 `inta[10]={1,2,3,4,5,6,7,8,9,10}, *p=a;`

则数值为 9 的表达式是: ( B. `*(p+8)` )。

201、有语句: `inta[10];` 则 ( ) 是对指针变量 `p` 的正确定义和初始化。--> B. `int *p=a;`

202、与结构成员访问表达式 `p->name` 等价的表达式为 ( )

B. `(*p).name`

203、与结构成员访问表达式 `x.name` 等价的表达式为 ( )。--> C. `(.&x)`

-name

204、语句 `int *p;` 说明了 ( )。--> C. `p` 是指向 `int` 型数据的指针

205、运算符优先级最高的是 ( ) A. ( )

206、运算符优先级最高的是 ( ) A. []

207、运算符优先级最高的是 ( )。--> B. +

208、运算符优先级最高的是 ( )。--> D. !=

209、在 C 语言程序中，表达式 `5%2` 的结果是: ( )。--> C. 1

210、在 C 语言程序中，单行或行尾注释语句使用的标记符为 ( )。

--> C. //

211、在 C 语言程序中，多行注释语句使用的开始标记符为 ( )。

--> B. /\*

212、在 C 语言中，表示换行符的转义字符为 ( )。--> C. \n

213、在 C 语言中，打开一个数据文件的系统函数为 ( )。 A. `fopen` ( )

214、在 C 语言中，每条复合语句的开始标记字符为 ( )。--> C. {

215、在 C 语言中，每条复合语句的结束标记字符为 ( )。 D. }

216、在 C 语言中，所有预处理命令都是以 ( ) 符号开头的。--> B. #

217、在 C 语言中，为只写操作打开一个文本文件的方式为 ( )。

--> D. "w"

218、在 C 语言中，一条简单语句的结束符是 ( ) C. 分号

219、在 C 语言中，字符型数据在计算机内存中，以字符的 ( ) 形式存储。--> C. ASCII 码

220、在 for 循环语句“for (i=0; i<n; i++) S;”中, 循环体 S 被执行的次数为 ( )。-->A.n

221、在 printf ( ) 函数调用的格式字符串中, 每个格式符的先导字符为 ( ) B.%

222、在 printf ( ) 函数调用的格式字符串中, 若使用格式符“%5d”, 则规定对应输出的数据占用的字符位置个数为 ( )。-->A. 5

223、在 printf ( ) 函数调用的格式字符串中, 若使用格式符“%c”, 则对应输出的数据类型为 ( )。-->A. char

224、在 printf ( ) 函数调用的格式字符串中, 若使用格式符“%d”, 则对应输出的数据类型为 ( )。-->B.int

225、在 scanf ( ) 函数调用的格式字符串中, 每个格式符的先导字符为 ( )。B.%

226、在 switch 语句的每个 case 块中, 假定都是以 eak 语句结束的, 则此 switch 语句容易被改写成的语句是 ( )。-->B. if

227、在程序的一个文件中定义的函数, 若要在另一个文件中调用, 则必须在这另一个文件中给出该函数的 ( )。-->A. 原型语句

228、在函数调用时, 以下说法正确的是: ( )。-->B. 实际参数和形式参数可以同名

229、在结构类型的定义中, 不同数据成员的定义项之间采用的分隔符是 ( )。-->C. 分号

230、在每个 C 语言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

231、在下列的符号常量定义中, 错误的定义语句格式为 ( )。-->C. constintM310;

232、在下列的符号常量定义中, 正确的定义格式为 ( )  
C. #define M3 10

233、在下列的字符数组定义中, 存在语法错误的是 ( )。  
D.chara[10]=5;

234、在下列选项中, 不正确的函数原型格式为 ( )。-->C. intFunction (a);

235、在下面的 do-while 循环语句中, 其循环体语句被执行的次数为 ( ), inti=0; do1++; while (i<10)。-->C.10

236、在下面的 do-while 循环语句中, 其循环体语句被执行的次数为 ( )。-->C. 10

237、在下面的 do 循环语句中, 其循环体被执行的次数为 ( )。  
A.4

238、在下面的 do 循环语句中, 其循环体被执行的次数为 ( )。inti=0; doi++; while (i\*i<10); -->4

239、在下面的保留字中, 不能作为函数的返回值类型的是 ( )。-->C. enum

240、在下面的二维数组定义中, 语法正确的选项是 ( )。-->C. inta[ ][3]={{1,3,5},{2}};

241、在下面的函数声明语句中, 存在着语法错误的是 ( )。-->C. AA (inta; intb)。

242、在下面的函数声明语句中存在着语法错误的是 ( )。-->C. AA (inta; intb)。

243、在下面的选项中, 不能作为函数的返回值类型的是 ( )。-->B.node

244、在下面的二维数组定义中, 错误的定义格式为 ( )。-->C. inta[ ]

245、在下面的一维数组定义中, 错误的定义为: ( )。-->C. chara[10]={0};

246、在下面的一维数组定义中, 有语法错误的选项是 ( )。-->C. inta[ ];

247、在下面循环语句中, 内层循环体 S 语句的执行总次数为 ( )。for(inti=0;i<n;i++) for(intj=i;j<n;j++)S; 答案: D.n(n+1)/2

248、在循环语句“for (i=0, i<n;i++) S, “中, 假定 n 的整数值大于 0, 则循环体 S 被执行的次数为 ( )。-->C.n

249、在循环语句“for (i=0; i<n; i++) S; “中, 循环体 S 被执行的次数为 ( ) C.n

250、在循环语句“for (i=1; i<10; i+=2) S;”中, 循环体 S 被执行的次数为 ( ) D.5

251、在循环语句“for (i=1; i<10; i+=3) S;”中, 循环体 S 被执行的次数为 ( ) A.3

252、在循环语句“for (i=n-1; i>=1; i--) S; “中, 循环体被执行的次数为 ( )。B. n-1

253、在一个程序文件中, 若要使用#include 命令包含一个系统头文件, 则此头文件所使用的起止定界符为一对 ( )。A.尖括号

254、在一个程序文件中, 若要使用#include 命令包含一个用户定义的头文件, 则此头文件所使用的起止定界符为一对 ( )。-->B. 双引号

255、在一个函数定义中, 函数头包括的 3 个部分依次为函数类型、函数名和 ( )。-->C. 参数表

256、在一个数组中, 数组元素 a[i] 前面的元素个数为 ( )。-->A.i

257、转义字符“\”表示的字符是 ( )。-->C. 反斜线

258、字符串“a+b=12/n”的长度为 ( )。-->B. 7

259、字符串“y=%d/n”的长度为 ( )。-->A.5

判断(223)--电大资源网: <http://www.dda123.cn/>(微信搜: 905080280)

1、continue 只能使用在任一种循环语句的循环体中。对

2、C 语言系统中提供一个用来描述文件属性的类型标识符为 FILE。-->对

3、C 语言中, 数组的下标是从 0 开始的。-->对

4、C 语言中, 转义字符“\n”的功能是换行。-->对

5、C 语言中的变量名只能由字母、数字和下划线三种字符组成, 且第一个字符必须为字母或者下划线。-->对

6、C 语言中的标准输入和输出设备文件都属于字符文件。-->对

7、C 语言中的每条复合语句以花括号作为结束符。-->错

8、C 语言中的每条简单语句以分号作为结束符。-->对

9、C 语言中的数据文件只包括字符文件这一种存储类型。-->错

10、do-while 循环语句的循环体, 可以是简单语句, 也可以是复合语句。正确

11、do-while 循环语句的循环体, 只能是复合语句, 不能是简单语句。 ( )。-->错

12、float 的类型长度为 8。-->错

13、for 循环语句的头部, for 后面圆括号内共有个表达式位置。 ( )。-->错

14、NULL 是一个符号常量, 通常作为空指针值, 它代表的值为 0。-->对

15、return 语句只有保留字, 没有其他成分。 ( ) -->错

16、return 语句中可以带有返回表达式。 ( )。-->对

17、strcmp 函数用于进行两个字符串之间的比较。-->对

18、strcpy 函数用于把一个字符串拷贝到另一个字符数组空间中。-->对

19、while 循环是先判断循环条件, 当条件为真时执行循环体。-->对

20、while 循环是先执行循环体, 然后进行循环条件判断。-->错

21、表达式 1%5 的值为 5。-->错

22、表达式 21%3 的值为 0。-->对

23、表达式 45%13 的值为 3。-->错

24、表达式 x=x+1 表示成增量表达式为 ++x。-->对

25、表达式 x=x+1 表示成增量表达式为 x++。 ( ) -->错

26、表达式 x=x+y 表示成复合赋值表达式为 x+=Yo ( )。-->对

27、表达式 x=x+y 表示成复合赋值表达式为 x+=y。错误

28、表达式 x=x-1 表示成减量表达式为 --x。 ( )。-->对

29、表达式 (float) 25/4 的值为 60 ( )。-->错

30、表达式 (float) 25/4 的值为 6。-->错

31、表达式 (int) 14.6%5 的值为 4。 ( ) 对

32、常数 3.26 是双精度定点数。 ( )。-->对

33、常数 3. 26f 是单精度浮点数。 ( )。-->错

34、存储一个空字符串需要占用 0 个字节的存储空间。-->错

35、存储字符'a'需要占用 1 个字节的存储空间。-->对

36、带有随机函数的表达式 rand ( ) %20 的值所在的区间范围是 0-19。-->对

37、当不需要函数返回任何值时, 则需要使用保留字 void 作为函数的返回值类型。-->对

38、当向字符文件输出一个换行符时, 实际将输出的是回车符。-->错

39、当向字符文件输出一个换行符时, 实际将输出的是回车符或换行符。 ( )。-->错

40、当循环次数确定时, while 循环语句能够转换为 for 循环语句。 ( )。-->对

41、调用系统函数时, 要先使用#include 命令包含该系统函数的原型语句所在的系统头文件。-->对

42、定点数既有双精度表示, 又有单精度表示。对

43、定义结构类型的变量时, 不能够同时进行初始化。-->错

44、定义结构类型的变量时, 能够同时进行初始化。-->对

45、定义结构类型的变量时, 能够用同类型的其他变量进行初始化, 不能用数据表进行初始化。错误

46、定义结构类型的变量时, 能够用同类型的其他变量进行初始化。 ( )。-->对

47、定义结构类型的同时能够定义它的变量。-->对

48、对二进制文件进行读和写操作将使用不同的系统函数。-->对

49、对二进制文件进行读和写操作将使用相同的系统函数。-->错

50、对于标识为 stati-->错

51、对于一个长度为 n 的字符串, 存储它的字符数组的长度至少为 n。错误

52、访问结构成员访问运算符只有一种方式, 即使用点运算符。错

53、关系表达式 x+y>5 的相反表达式为 x+y<=5-->对

54、关系表达式 (x !=0) 的等价表达式为 (!x)。错

- 55、关系表达式  $(x!=0)$  的等价表达式为  $(x)$ 。-->**对**
- 56、关系表达式  $(x! =0)$  的等价表达式为  $(! x)$ 。-->**错**
- 57、函数定义格式中的参数表被称为实参表。()。-->**错**
- 58、函数形参变量不属于局部变量。-->**错**
- 59、假定 a 是一个一维数组, 则 a+i 的值与 &a[i] 的值相等。-->**对**
- 60、假定 a 是一个一维数组, 则表达式  $*(a+i)$  所对应的元素为  $a[i - 1]$ 。()。-->**错**
- 61、假定 a 是一个一维数组, 则进行  $a++$  运算是允许的。() -->**对**
- 62、假定 a 是一个指针数组, 则 a+i 所指对象的地址比 a 地址大  $4*i$  字节。-->**对**
- 63、假定 p 所指对象的值为 25, p+1 所指对象的值为 46, 则  $*++p$  的值为 25。-->**错**
- 64、假定 p 所指对象的值为 25, p+1 所指对象的值为 46, 则  $*p++$  的值为 46。-->**错**
- 65、假定 p 所指对象的值为 25, p+1 所指对象的值为 46, 则执行  $*(p++)$  运算后, p 所指对象的值为 46。-->**对**
- 66、假定 p 所指对象的值为 25, p+1 所指对象的值为 46, 则执行  $*(p++)$  运算后, p 所指对象的值为 46。() -->**对**
- 67、假定  $x=5$ , 则表达式  $2*x++$  的值为 12。-->**错**
- 68、假定  $x=5$ , 则执行  $"a=(x? 10:++x)"$  语句后 a 的值为 10。-->**对**
- 69、假定  $x=5$ , 则执行  $"a=(x? 10: 20)"$  语句后 a 的值为 10。-->**对**
- 70、假定  $x=5$ , 则执行  $"y=++x-->对$
- 71、假定  $x=5$ , 则执行  $"y=++x;"$  语句后, y 的值为 6。-->**对**
- 72、假定  $x=5$ , 则执行  $"y=x++B;"$  语句后, x 的值为 5。-->**错**
- 73、假定 x 为一个简单变量, 则 &x 表示 x 的地址。-->**对**
- 74、假定二维数组的定义为  $"chara[M][N]B"$ , 则该数组所含元素的个数为  $M+N$ 。-->**错**
- 75、假定二维数组的定义为  $"doublea[M][N];"$ , 则每个数组元素的列下标取值范围在  $0\sim N$  之间。-->**错**
- 76、假定二维数组的定义为  $"doublea[M][N];"$ , 则每个数组元素的行下标取值范围在  $0\sim M-1$  之间。-->**对**
- 77、假定二维数组的定义为  $"inta[3][5]B"$ , 则该数组所占存储空间的字节数为 60。-->**对**
- 78、假定二维数组的定义语句为  $"doublea[M][N];"$ , 则每个数组元素的列下标取值范围在  $0\sim N-1$  之间。()。-->**对**
- 79、假定是一个一维数组, 则表达式  $*Ca+i)$  所对应的元素为  $[\ ]$ 。()。-->**对**
- 80、假定是一个一维数组, 则进行  $a++$  运算是允许的。()。-->**对**
- 81、假定访问一个结构对象 x 中的由 a 指针成员所指的对象, 则表示方法为  $x.a$ 。-->**错**
- 82、假定访问一个结构指针 p 所指对象中的 b 指针成员所指的对象, 则表示方法为  $p->b$ 。-->**错**
- 83、假定一个结构类型的定义为  $"structA (inta,bBA*cB) B"$ , 则该类型的长度为 12。-->**对**
- 84、假定一个结构类型的定义为  $"structB (inta[5]Bchar*bB) B"$ , 则该类型的长度为 20。-->**错**
- 85、假定一个枚举类型的定义为  $"enumRB {ab,ac=3,ad,ae}x=ad;"$ , 则 x 的值为 2。-->**错**
- 86、假定一个数据对象为 inH 类型, 则指向该对象的指针类型为 inH 类型。()。-->**对**
- 87、假定一个数据对象为 int\* 类型, 则指向该对象的指针类型仍为 int\* 类型。-->**错**
- 88、假定一个数据对象为 int\* 类型, 则指向该对象的指针类型为 int\*\*。正确
- 89、假定一个数据对象为 int\* 类型, 则指向该对象的指针类型仍为 int\* 类型。() -->**错**
- 90、假定一维数组的定义为  $"char*a[8]B"$ , 则该数组所含元素的个数大于 8。-->**错**
- 91、进行动态存储分配的 calloc () 函数带有两个参数。()。-->**对**
- 92、进行动态存储分配的 malloc () 函数带有两个参数。()。-->**错**
- 93、进行动态存储分配的 malloc () 函数带有一个参数。-->**对**
- 94、空字符串的长度为 1。-->**错**
- 95、逻辑表达式  $(a>bb==5)$  的相反表达式为  $(a>b&&bb==5)$ 。-->**错**
- 96、每次只能向文本文件中写入一个字符, 不能一次写入一个字符串。-->**错**
- 97、每个 C 程序文件在编译时可能出现有警告性错误, 其应的标识符为 error。-->**错**
- 98、如果需要从被调用函数返回一个函数值, 被调用函数必须包含 return 语句。-->**对**
- 99、如果一个函数只允许同一程序文件中的函数调用, 则不应在该函数定义的开始前加上保留字 static。-->**错**
- 100、如果在一个函数体中又出现对自身函数的调用, 此种函数调用被称为递归调用。-->**对**
- 101、若 p 指向 x, 则 \*p 与 x 的值不同。() -->**错**
- 102、若  $x=15, y=10$ , 则  $x!=y$  的逻辑值为真。对
- 103、若  $x=15, y=20$ , 则  $x>y$  的逻辑值为真。-->**错**
- 104、若  $x=5, y=10$ , 则  $x>y$  逻辑值为真。-->**错**
- 105、若  $x=5, y=10$ , 则  $x<=y$  的逻辑值为假。()。-->**错**
- 106、若  $x=5, y=10$ , 则  $x<=y$  的逻辑值为真。正确
- 107、若需要定义一个符号常量, 并且使 C 语言能够进行类型检查, 则应在定义语句的开始使用保留字 const。-->**对**
- 108、若要把一个整型指针 p 转换为字符指针, 则采用的强制转换表达式为  $(char*) p$ 。-->**对**
- 109、若要在程序文件中使用标准输入和输出函数, 则需要引入的系统头文件为 stdio。()。-->**对**
- 110、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数组首地址。-->**对**
- 111、若有定义:  $doublex[3][5];$  则数组中列下标的上限为 4。-->**对**
- 112、设  $a=3b=2, c=1$ , 则表达式  $f=a>b>c$  的值是 0。-->**对**
- 113、设 a 为 int 型变量, 执行下列赋值语句  $a=125. 534;$  后, a 的取值是 125。-->**对**
- 114、设  $inta=12$ , 则执行完语句  $a=++a*a$  后, a 的值是 0。-->**对**
- 115、十进制数 25 表示成符合 C 语言规则的八进制数为 31。-->**错**
- 116、十进制数 25 表示成符合 C 语言规则的十六进制数为 0x19。-->**对**
- 117、使用  $"typedefcharBB[10][50]B"$  语句定义标识符 BB 为含有 10 行 50 列的二维字符数组类型。-->**对**
- 118、使用 const 语句定义一个符号常量时, 不必对它进行初始化。-->**错**
- 119、使用 const 语句定义一个符号常量时, 不需要对它同时进行初始化。()。-->**错**
- 120、使用 const 语句定义一个符号常量时, 则必须对它同时进行初始化。() -->**对**
- 121、使用一个结构类型时, 必须一起使用关键字 struct 和类型标识符。-->**对**
- 122、为了存储一个长度为 n 的字符串, 所使用的字符数组的长度至少为  $n+1$ 。-->**对**
- 123、为了结束本层循环类语句或 switch 语句的执行过程, 在语句体中需要使用 break 语句。-->**对**
- 124、系统默认的 C 语言源程序文件的扩展名是 .cpp。-->**错**
- 125、一个 C 程序可以有多个 main 函数。-->**错**
- 126、一个 C 语言程序能够包含多个用户头文件。-->**对**
- 127、一个 C 语言程序只能包含一个用户头文件。错
- 128、一个磁盘数据文件的文件名由文件主名和扩展名所组成, 其中间用圆点分开。-->**对**
- 129、一个二维字符数组  $a[10][20]$  能够存储的每个字符串的长度不超过 20。-->**错**
- 130、一个二维字符数组  $a[10][20]$  能够最多存储 9 个字符串。-->**错**
- 131、一个二维字符数组  $a[10][20]$  中存储每个字符串的最大长度为 190。()。-->**对**
- 132、一个二维字符数组  $a[10][20]$  中存储每个字符串的最大长度为 200。()。-->**错**
- 133、一个二维字符数组  $a[10][20]$  中存储每个字符串的最大长度为 20。() -->**错**
- 134、一个数组名是一个指针常量, 其值可以被修改。()。-->**错**
- 135、已知  $x=-25$ , fabs (x) 的值为 -25。()。-->**错**
- 136、已知  $x=-25$ , 则 fabs (x) 的值为 -25.0。() -->**错**
- 137、已知  $"inta[10], *p=a;"$ , 则进行  $p++$  运算是允许的。对
- 138、已知函数定义为:  $intstat (inta, float*p)$ , 则此函数的 2 个参数类型分别是整型、指针型。-->**对**
- 139、已知字符 'C' 的 ASCII 码为 67, 当执行  $"intx='C'+5-->对$
- 140、已知字符 'C' 的 ASCII 码为 67, 当执行  $"intx='C'+5;"$  语句后 x 的值为 72。-->**对**
- 141、用于输出表达式值的标准输出函数是 printf ()。-->**对**
- 142、有如下函数头:  $intmax (inti, intj)$  则此函数有 2 个参数。-->**对**
- 143、有一条语句为  $"chars2[4]="abcd";"$ , s2 数组长度的定义是合适的。错
- 144、与结构成员访问表达式  $(*fp).score$  等价的表达式是  $fp->score$ 。-->**对**
- 145、语言程序中, 对于递归函数和非递归函数, 其函数头部有区别标志。()。-->**错**

- 146、语言程序中，只有一种使用注释语句的方法。()。-->**错**
- 147、语言程序中，主函数也可以成为递归函数。()。-->**错**
- 148、语言中，在定义一个字符数组时，不能利用一个字符串进行初始化。()。-->**错**
- 149、源程序文件被编译成目标文件后，其目标文件中也存在有对应的注释内容。()。-->**错**
- 150、在 C 语言程序上机操作的过程中，编译程序的操作步骤之后是连接操作。-->**对**
- 151、在 C 语言程序上机操作的过程中，编译程序的操作步骤之后是运行操作。()。-->**错**
- 152、在 C 语言程序中，RBC 和 Rbe 可以被作为同一个标识符使用。-->**错**
- 153、在 C 语言程序中，存储字符串的方法是使用字符数组。对
- 154、在 C 语言程序中，存储字符串的方法是使用字符指针。-->**错**
- 155、在 C 语言程序中，对于递归函数和非递归函数，其函数头部无区别标志。对
- 156、在 C 语言程序中，对于递归函数和非递归函数，其函数头部有区别标志。-->**错**
- 157、在 C 语言程序中，有两种使用注释语句的方法。对
- 158、在 C 语言程序中，在行尾使用注释的开始标记符为一单斜线字符。-->**对**
- 159、在 C 语言程序中，只有一种使用注释语句的方法。() -->**错**
- 160、在 C 语言程序中，主函数可以是递归函数。-->**错**
- 161、在 C 语言中，保留字是有专门含义和作用的，不能作为一般标识符使用。()。-->**对**
- 162、在 C 语言中，常数 28 和 326 具有相同的数据类型。-->**错**
- 163、在 C 语言中，对二进制文件的所有不同打开方式共有 6 种。-->**对**
- 164、在 C 语言中，逻辑运算符“||”称为逻辑或。-->**对**
- 165、在 C 语言中，每个保留字是有专门含义和作用的，不能作为一般标识符使用。() -->**对**
- 166、在 C 语言中，系统函数 fseek () 用来移动数据文件中的文件位置指针。-->**对**
- 167、在 C 语言中，一个函数由函数头和函数体组成。-->**对**
- 168、在 C 语言中，一个结构类型的长度等于所含的所有数据成员类型的长度之和。()。-->**对**
- 169、在 C 语言中，以 \n 作为字符串结束标志。-->**错**
- 170、在 C 语言中，在定义一个指针变量时，不能对其进行初始化。-->**错**
- 171、在 C 语言中，在定义一个字符数组时，不能利用一个字符串进行初始化。错
- 172、在 C 语言中，在定义一个字符数组时，可以同时利用一个字符串进行其初始化。() -->**对**
- 173、在 C 语言中，转义字符 \n 表示一个换行符。-->**对**
- 174、在 C 语言字符集中，包含有全部 26 个英文大写字母和对应的小写字母。对
- 175、在 C 语言字符集中，一个大写英文字母和它的小写英文字母被视为不同的字符。()。-->**对**
- 176、在 for 循环语句的头部，for 后面圆括号内共有 3 个表达式位置。() -->**对**
- 177、在 switch 语句中，每个 case 和冒号之间的表达式只能是常量。-->**对**
- 178、在程序执行完成任一个函数调用后，将结束整个程序的执行过程，返回到操作系统或 C 语言集成开发环境界面窗口。-->**错**
- 179、在程序中执行到 break 语句时，将结束所在函数的执行过程，返回到调用该函数的位置。-->**错**
- 180、在单行或多行注释中，其在释内容前需要使用标记符“j 养”，而在结束处不需要使用“\*/”。() -->**错**
- 181、在定义一个变量时，不能给它赋初值。-->**错**
- 182、在定义一个变量时，不能够同时给它赋初值。()。-->**错**
- 183、在定义一个变量时，可以同时给它赋初值。-->**对**
- 184、在定义一个字符数组时，可以同时利用一个字符串进行其初始化。-->**对**
- 185、在定义指针变量的语句“int \*p, pp;”中，p 和 pp 具有不同的数据类型。对
- 186、在二维数组的定义语句中，不能给数组中每个元素赋初值。()。-->**错**
- 187、在二维数组的定义语句中，可以给数组中每个元素赋初值。对
- 188、在二维数组的定义语句中，数组名后带有两对圆括号。() -->**错**
- 189、在二维数组的定义语句中，数组名前为元素类型关键字。()。-->**对**
- 190、在函数模块之外定义的变量称为全局变量，若没有被初始化则系统隐含对它赋初值 0。-->**对**
- 191、在结构成员访问运算符中，点运算符(.)为双目运算符。()。-->**对**
- 192、在结构成员访问运算符中，点运算符和箭头运算符的左边均为结构变量。()。-->**错**
- 193、在结构成员访问运算符中，点运算符和箭头运算符的左边均为结构指针变量。()。-->**错**
- 194、在结构类型的定义中，结构类型的作用域范围，与它的定义位置有关。()。-->**对**
- 195、在结构类型的定义中，结构类型名在关键字 struct 和定义体之间。() 对
- 196、在结构类型的定义中，其中的数据成员可以是本身类型的直接对象。-->**错**
- 197、在结构类型的定义中，允许出现结构类型不完整的超前定义。-->**对**
- 198、在数据文件打开方式字符串中，字符 r、w 和 a 具有确定的含义，分别代表读、写和追加方式。-->**对**
- 199、在同一个结构类型的定义中，各数据成员名必须不同。()。-->**对**
- 200、在一个 C 语言程序文件中，若要包含一个头文件，则使用以百分号开始的预处理命令。-->**错**
- 201、在一个磁盘数据文件的文件名中，文件主名和扩展名都是必须的，不可省略。-->**错**
- 202、在一个磁盘数据文件的文件名中，文件主名是必须的，扩展名可任选择使用。()。-->**对**
- 203、在一个函数定义中，参数表中的参数说明项之间是采用逗号分开的。-->**对**
- 204、在一个函数定义中，函数体是一条复合语句。-->**对**
- 205、在一个函数定义中，函数体是一条简单语句。()。-->**错**
- 206、在一个函数定义中，函数体是用一对花括号括起来的。()。-->**对**
- 207、在一个链表的结点结构中，必然包含有一个指向自身结点类型的指针域。对
- 208、在一条变量定义语句中，定义多个变量时需要使用分号隔开。错误
- 209、在一条变量定义语句中，可以同时定义不同基准类型的变量。()。-->**错**
- 210、在一维数组的定义语句中，数组名后带有一对花括号。错
- 211、在一维数组的定义语句中，数组名后带有一对圆括号。对
- 212、在一维数组的定义语句中，数组名后使用中括号把数组长度值括起来。-->**对**
- 213、在一维数组的定义语句中，数组名前为元素类型关键字。()。-->**对**
- 214、在执行“typedef int DataTypeB”语句后，标识符 DataType 与保留字 int 具有完全相同的含义和作用。-->**对**
- 215、执行“printf(“%c,F’-2) -->**错**
- 216、执行“printf(“%c”,F’-2);”语句后得到的输出结果为 H。-->**错**
- 217、执行 calloc (n,8) 函数调用时，将动态分配得到 8\*n 个字节的存储空间。-->**对**
- 218、执行 malloc (sizeof (struct BB)) 函数调用时，得到的动态存储空间能够保存具有 struct BB 结构类型的一个对象。-->**对**
- 219、注释内容的开始所使用的标记符为/\*，则注释内容的结束所使用的标记符为\*/。-->**对**
- 220、字符串 aA//xxk//数据的长度为 13。-->**错**
- 221、字符串“a: xxk 数据”的长度为 13。-->**错**
- 222、字符串不允许为空，至少需要包含一个字符。错
- 223、字符串允许为空，并且其长度为。()。-->**对**
- 填空(129)--电大资源网: <http://www.dda123.cn/> (微信搜: 905080280)
- 1、C 程序中的所有预处理命令均以 () 字符开头。-->**#(或井号)。**
  - 2、C 语言中的每个程序文件在编译时可能出现有 error 类型的错误，它属于 (致命性 / 警告)
  - 3、C 语言中的每条复合语句以 () 作为结束符。-->**《或右花括号》。**
  - 4、C 语言中的每条复合语句以右 () 括号作为结束符。-->**花()。**
  - 5、C 语言中的每条简单语句以 () 作为结束符。-->**;(或分号)。**  
**C 程序中的所有预处理命令均以 () 字符开头。答案: #(或井号)。**
  - 6、double 类型的长度为 ()。-->**8**
  - 7、int 数据类型的长度为 ()。-->**4**
  - 8、NULL 是一个符号常量，作为空指针，它所表示的整数值为 ()。-->**0**
  - 9、shortint 类型的长度为 ()。-->**2**
  - 10、strcat () 函数用于 () 两个字符串。-->**连接**
  - 11、strcpy 函数用于把一个字符串 () 到另一个字符数组空间中。-->**拷贝(复制)。**

- 12、strlen () 函数用于计算一个字符串的 ( )。-->长度
- 13、treat () 函数用于 ( ) 两个字符串。-->连接
- 14、把表达式  $x=x+y$  转换成复合赋值表达式为 ( )。--> $x+=Y$
- 15、把间接访问表达式  $p->name$  改写成对应的直接访问表达式为 ( )。-->( \*p ) . name
- 16、保留字 case 和 default 出现在 (if/switch/for/while) ( ) 语句中。-->switch
- 17、表达式  $25/4$  的值为 ( )。-->6
- 18、表达式 (float)  $25/4$  的值为 6.25。
- 19、表达式 (int)  $1-1$  为 14
- 20、程序中的每个编译单位是一个 ( )。-->程序文件
- 21、存储一个空字符串需要占用 ( ) 个字节。-->1
- 22、存储字符 'a' 需要占用存储器的 ( ) 个字节空间。-->1
- 23、存储字符串 "a" 需要至少占用存储空间中的 ( ) 个字节。-->2
- 24、当不需要函数返回任何值时, 则应使用 ( ) 保留字来定义函数类型。-->void
- 25、对于 "for (i=0; i < 10; i+=2) S" 语句, 循环体 S 被执行的次数为 ( )。-->5
- 26、对于 "for (i=0; i < 10; i++) S" 语句, 循环体 S 被执行的次数为 ( )。-->10
- 27、关系表达式  $a > b$  的相反关系式为 ( )。--> $a <= b$
- 28、很定要动态分配一个类型为 struct worker 的对象, 并由 r 指针指向这个对象, 则 r 的数据类型为 ( )。-->struct Worker\*
- 29、假定一个枚举类型的定义为 "enum RA (ab, ad, ae) 户, 则 ad 的值为 ( )。-->2
- 30、假定 a 是一个一维数组, 则 a[i] 的指针访问方式为 ( )。-->\*(a+i)
- 31、假定 p 是一个变量, 则该变量的地址表示为 ( )。-->&p
- 32、假定 p 是一个指向整数对象的指针, 则用 ( ) 表示该整数对象。-->\*p
- 33、假定 p 所指对象的值为 2-->25
- 34、假定 p 所指对象的值为 25, p+1 所指对象的值为 46, 则 'p+1' 的值为 ( )。-->25
- 35、假定 P 所指对象的值为 25, p+1 所指对象的值为 46, 则 \*++p 的值为 ( )。-->46
- 36、假定 p 所指对象的值为 25, p+1 所指对象的值为 46, 则执行表达式 (\*p)++ 后, p 所指对象的值为 ( )。-->26
- 37、假定 p 所指对象的值为 25, p+1 所指对象的值为 46, 则执行表达式 (P++) 后, p 所指对象的值为 ( )。-->46
- 38、假定 p 所指对象的值为 25, p+1 所指对象的值为 46, 则 \*++p 的值为 46。
- 39、假定 p 所指对象的值为 25, p+1 所指对象的值为 46, 则洛 p++ 的值为 ( )。-->25
- 40、假定 p 所指对象的值为 25, p+1 所指对象的值为 46, 则执行表达式 (\*p)++ 后, p 所指对象的值为 26。
- 41、假定 P 所指对象的值为 34, p-1-1 所指对象的值为 4-->34
- 42、假定  $x=5$ , 则执行  $a=(x? 10: 20)$ ; "语句后 a 的值为 ( )。-->10
- 43、假定  $x=5$ , 则表达式  $2+x++$  的值为 7。
- 44、假定  $x=5$ , 则表达式  $2+x++$  的值为 ( )。-->7
- 45、假定  $x=5$ , 则执行 "a=(!x? 10: 20); " 语句后 a 的值为 20。
- 46、假定  $x=8$ , 则表达式  $2+x++$  的值为 ( )。-->10

- 47、假定 x 的值 5, 则执行 "a=(!x? 10: 20); " 语句后 a 的值为 ( )。-->20
- 48、假定 x 的值为 -->8
- 49、假定 x 的值为 5, 则执行 "a=(!x? 10: 20); " 语句后 a 的值为 (20)。
- 50、假定  $y=10$ , 则表达式  $++y*3$  的值为 ( )。-->33
- 51、假定不允许使用逻辑非操作符, 则关系表达式  $x+y > 5$  的相反表达式为 ( )。--> $x+y <= 5$
- 52、假定二维数组的定义为

假定二维数组的定义为 "double a[M][N];", 则所有元素的列下标的取  
间。  
答案:  $0-N-1$

- 53、假定二维数组的定义为 "char a[M][N];", 则数组共包含有 ( ) 个字符元素。--> $M*N$
- 54、假定二维数组的定义为 "double a[M][N];", 则每个数组元素的行下标取值范围在  $0-M-1$  之间。
- 55、假定二维数组的定义为 "double a[M][N];", 则数组元素的行下标取值范围在 ( ) 之间。--> $0-M-1$
- 56、假定二维数组的定义为 "double a[M][N];", 则所含元素的列下标的最大值为 ( )。--> $N-1$
- 57、假定二维数组的定义为 "double a[M][N];", 则数组元素的行下标的最大值为 ( )。--> $M-1$
- 58、假定二维数组的定义为 "inta[3][5];", 则数组所包含的元素个数为 ( )。-->15
- 59、假定二维数组的定义为 "inta[3][5];", 则数组所占存储空间的字节数为 ( )。-->60
- 60、假定二维数组的定义为, "inta[3][5];", 则数组所包含的元素个数为 ( )。-->15
- 61、假定要动态分配一个类型为 struct Worker 的对象, 并由 r 指针指向这个对象, 则 r 的数据类型为 ( )。-->struct Worker\*
- 62、假定要动态分配一个类型为 struct Worker 的对象, 并由 r 指针指向这个对象, 则使用的表达式为 ( ) = malloc < size of (struct Worker) )。-->rker\*r
- 63、假定要访问一个结构对象 x 中的数据成员 a, 则表示方式为 ( )。-->x.a
- 64、假定一个结构类型的定义为 "struct A (inta, b; A\*c; );", 则该类型的理论长度为 ( )。-->12
- 65、假定一个结构类型的定义为 "struct B (inta[5]; char\*b; );", 则该类型的理论长度为 ( )。-->24
- 66、假定一个枚举类型的定义为 则 ad 的值为 ( )  
假定一个枚举类型的定义为 则 ad 的值为 ( )  
答案: 2
- 67、假定一个枚举类型的定义为 "enum RA (ab, a-->1
- 68、假定一个枚举类型的定义为 "enum RA {ab, ac, ad, ae};", 则 ac 的值为 (1)。
- 69、假定一个数据对象为 int\* 类型, 则指向该对象的指针类型为 ( )。-->int\*\*

- 70、假定一维数组的定义为 "Char\*a[8];", 则数组所占用的存储空间的字节数为 ( )。-->32
- 71、假定一维数组的定义为 "char\*a[M];", 则数组所占存储空间的字节数为 ( )。-->4\*M
- 72、假定一维数组的定义为 "char\*a[8];", 则数组所占用的存储空间的字节数为 ( )。-->32
- 73、假定一维字符指针数组的定义为 "Char\*a[5];", 则数组占用的存储空间的字节数-->20
- 74、假定一维字符指针数组的定义为 "Char\*a[8];", 则数组占用的存储空间的字节数为 ( )。-->32
- 75、空字符串的长度为 ( )。-->0
- 76、类型。-->BB
- 77、逻辑表达式  $(x==0 \& \& y > 5)$  的相反表达式为 ( )。--> $(x != 0 \vee y <= 5)$   
或:  $(xy <= 5)$
- 78、每个 C 语言程序文件在编译时可能出现有致命性错误, 其对应的标识符为 ( )。-->errorC.知'A'~'Z'的ASCII码为65-90.当执行 "int x=C+3; " 语句后 x 的值为 ( )。答案: 70
- 79、若 p 指向 x, 则 ( ) 与 x 的表示是等价的。-->\*p
- 80、若  $x=5, y=10$ , 则执行  
若  $x=5, y=10$ , 则执行 "if (x > y) c=5; else c=10; " 语句后, c 的值为 \_\_\_\_\_。  
答案: 10
- 81、若  $x=5, y=10$ , 则  $x!=y$  的逻辑值为 ( )。-->1 (真, true)。
- 82、若  $x=5, y=10$ , 则  $x > y$  的值为 ( )。-->0 (假)。
- 83、若  $x=5, y=10$ , 则执行 "if (x <= y) c=3; else c=8; " 语句后, c 的值为 ( )。-->3
- 84、若  $x=5, y=8$ , 则执行 "if (x > y) C=2; else C=3; " 语句后, -->3
- 85、若  $x=5, y=10$ , 则  $x > y$  的值为 0 (假)。
- 86、若要把一个整型指针 P 转换为字符指针, 则采用的强制转换表达式为 ( )。-->(char\*)。
- 87、若一个字符串的长度为 n, 则存储它的字符数组的长度至少为 ( )。-->n+1
- 88、十进制数 25 表示成符合 C 语言规则的十六进制数为 ( )。-->0x19
- 89、十进制数 35 对应的八进制数为 43。
- 90、十进制数 35 对应的十六进制数为 ( )。-->0x23
- 91、使用 "typedef char BB[10][50]; " 语句定义 BB 为含有 10 行 50 列的二维字符数组类型。
- 92、一个函数定义由和 ( ) 函数体两部分组成。-->函数头
- 93、一个空字符串的长度为 ( )。-->0
- 94、用类型关键字表示十进制常数-->float
- 95、用类型关键字表示十进制常数 326f 的类型为 float。
- 96、用于从键盘上为变量输入值的标准输入函数的函数名为 ( )。-->scanf
- 97、用于存储一个长度为 n 的字符串的字符数组的长度至少为 n+1。
- 98、用于输出表达式值的标准输出函数的函数名是 ( )。-->printf
- 99、与结构成员访问表达式 (\*fp).score 等价的表达式为 fp->score。
- 100、在 #include 命令中所包含的文件, 可以是头文件, 也可以是程序文件。

101、在#include 命令中所包含的文件,可以是头文件,也可以是  
( )文件。-->程序

102、在 C 语言中,一个函数定义由函数头和( )这两个部分组成。  
-->函数体

103、在程序中执行到 return 或返回语句时,将结束所在函数的执行过程,返回到调用该函数的位置。

104、在函数调用中执行到 (return/continue/if) ( ) 语句时,将结束所在函数的执行过程,返回到调用该函数的原来位置。  
-->return

105、在函数外定义的变量,若没有被初始化则系统隐含对它所赋的初值为 0。

106、在件 i, lud. 命令中所包含的文件.可以是头文件,也可以是  
( )文件。-->程序

107、在所有函数定义之外定义的变量,若没有被初始化则系统隐含对它所赋的初值为 ( )。-->0

108、在一个 C 语言程序文件中,若要包含另外一个头文件或程序文件,则此包含命令的标识符为 ( )。-->#include

109、在一个程序的函数之外定义的整型变量,若没有被初始化,则系统隐含对它所赋的初值为 ( )。-->0

110、增量表达式+y 表示成赋值表达式为-->y=y+1

111、增量类型 int 的长度为 ( )。-->4

112、执行“intx=45,y=13; printf (“%d”,x/y);”语句序列后得到的输出结果为 ( )。-->3

113、执行“intx=45,y=8; printf (“%d”,x/y);”语句序列后得到的输出结果为 ( )。-->5

114、执行“printf (“%c”,A+2);”语句后得到的输出结果为 ( )。  
-->C

115、执行“printf (“%c”,F-2);”语句后得到的输出结果为 ( )。  
-->D

116、执行“typedefintABC[10];”语句把 ABC 定义为具有 10 个整型元素的\_数组\_类型。

117、执行“typedefintABC[10];”语句把 ABC 定义为具有 10 个整型元素的 ( ) 类型。-->一维数组

118、执行“typedefintDataType;”语句后, ( ) 与 int 保留字具有相同作用。-->DataType

119、执行“typedefintabc[10];”语句把 AB-->数组

120、执行 int\*p=mallo (sizeof (int)) 操作得到的一个动态分配的整型对象为 ( )。-->\*p

121、执行“intx=45,y=13; printf (“%d”, x/y);”语句序列后得到的输出结果为 (3)。

122、执行“intx=45,y=8; printf (“%d”, x/y);”语句序列后得到的输出结果为 ( )。-->5

123、执行“typedefintABC[10];”语句把 ABC 定义为具有 10 个整型元素的一维 ( ) 类型。-->数组

124、执行“typedefintDataType;”语句后, ( ) 与 int 保留字具有相同作用。-->DataType

125、直接访问表达式 (\*fp).score 所对应的间接访问表达式为 (fp->score)。

126、直接访问表达式 (\*fp).score 所对应的间接访问表达式为 ( )。  
-->fp -->score

127、字符串“a: //xxk//file.txt”的长度为 ( )。-->15

128、字符串“a: //xxk//file.txt”的长度为\_15\_。

129、作为语句标号使用的 case 和 default 只能用于 switch 或开关或分情况\_语句的定义体中。

编程题(39)--电大资源网: http://www.dda123.cn/ (微信搜: 905080280)

- 1、按照 “int FF(int a[], int n)” 的函数声...
- 2、编写一个程序, 计算 1+3+32+...+310 的值并输出...
- 3、编写一个程序, 输出 50 以内 (含 50) 的、能够被 3 或...
- 4、编写一个递归函数“int FF(int a[], int n)...
- 5、编写一个递归函数“int FF(int a[], int n)...
- 6、编写一个递归函数“int FF(int a[], int n)...
- 7、编写一个函数, 函数头格式为“double Mean(do...
- 8、编写一个函数, 函数原型为“int fun4(char\*a)...
- 9、编写一个主函数, 计算 1+3+32+...+310 的值并输...
- 10、编写一个主函数, 计算并输出 12+22+...+n2 值, 其...
- 11、编写一个主函数, 假定 a 和 b 的取值范围为: ...
- 12、编写一个主函数, 利用 while 循环, 求出并显示满...
- 13、编写一个主函数, 利用 while 循环计算...
- 14、编写一个主函数, 求出满足不等式 22+42+...+n2...
- 15、补充完善下面的一个主函数, 假定函数体中定义...
- 16、补充完整下面 xzk2 函数定义, 假定函数参数表中...
- 17、补充完整下面的函数定义, 该函数要求返回满足...
- 18、补充完整下面的函数定义, 计算出 1+1\*3+2\*3+ ...
- 19、补充完整下面的函数定义, 计算出整型数组 a 中 n...
- 20、补充完整下面的函数定义, 利用 for 循环计算 1+2...
- 21、补充完整下面的函数定义, 利用 for 循环计算 1+4...
- 22、补充完整下面的函数定义, 要求返回二维数组 a[...
- 23、补充完整下面的函数定义, 要求返回二维数组 a[...
- 24、补充完整下面的函数定义, 要求返回二维数组 a[...
- 25、补充完整下面函数 xzk5 的定义, 计算并返回数组...
- 26、补充完整下面函数定义中 while 循环的循环体, ...
- 27、根据函数原型“double Mean(double a[M][N]...)...
- 28、根据函数原型“double Mean (doublea[M][N]i...)...
- 29、根据函数原型“doubleMean (doublea[M]ENI.in...)...
- 30、根据函数原型“int FF(int a[], int n)”, ...
- 31、根据函数原型“int FF(int a[], int n)”, 编...
- 32、根据函数原型“int MM(int a[], int m)”, 编写...
- 33、根据函数原型“int MM(int a[], intrn)”, 编...
- 34、根据函数原型“intDD (inta[], intrn,intMM)“编...
- 35、根据函数原型“intFF (inta[], int n)“, 编写函数...
- 36、根据函数原型“intMM (intaE],intrn)“, 编写函...
- 37、根据函数原型“void DD (int a[], int n, int MM) ...
- 38、完善下面程序, 该程序功能是从键盘上...
- 39、完善下面程序中 xzk1 函数的定义, 使程序能够输...

1、按照 “int FF(int a[], int n)” 的函数声明, 编写出递归函数的定义, 求出数组 a 中所有 n 个元素之积并返回。

```
int FF(int a[], int n)
{
    if(n<=0) {printf("n 值非法\n"),exit(1);}
    if(n==1) return a[0];
    else return a[n-1] * FF(a,n-1);
}
```

2、编写一个程序, 计算 1+3+32+...+310 的值并输出, 假定分别用 i,p,s 作为循环变量、累乘变量和累加变量的标识符。 程序为:

```
#include<stdio.h>
void main(){
    int i;
    int p=1;
    int s=1;
    for(i=1;i<=10;i++){p*=3; s+=p;}
    printf("%d\n",s); }
```

3、编写一个程序, 输出 50 以内 (含 50) 的、能够被 3 或者 5 整除的所有整数。  
程序为: #include<stdio.h>

```
void main() {
    int i;
    for(i=3; i<=50; i++)
        if(i%3==0 || i%5==0) printf("%d ",i);
    printf("\n"); }
```

4、编写一个递归函数“int FF(int a[], int n)”, 求出数组 a 中所有 n 个元素之积并返回。  
答: int FF(int a[], int n)

```
{
    if(n==1) return a[n-1];
    else return a[n-1]*FF(a,n-1);
}
```

5、编写一个递归函数“int FF(int a[], int n)”, 求出数组 a 中所有 n 个元素之积并返回。  
程序为: int FF(int a[], int n){

```
if(n<=0) {printf("n 值非法\n"),exit(1);}
if(n==1) return a[n-1];
else return a[n-1]*FF(a,n-1); }
```

6、编写一个递归函数“int FF(int a[], int n)”, 求出数组 a 中所有 n 个元素之积并返回。  
答案:

```
int FF(int a[], int n)
{
    if(n==1) return a[n-1];
    else return a[n-1]*FF(a,n-1);
}
```

7、编写一个函数，函数头格式为“double Mean(double a[M][N],int m,int n)”，要求返回二维数组 a[m][n]中所有元素的平均值，假定在计算过程中采用变量 v 存放平均值。

答：double Mean(double a[M][N],int m,int n)

```
{
    int i,j;
    double v=0.0;
    for(i=0; i<m; i++)
        for(j=0; j<n; j++) v+=a[i][j];
    v/=m*n;
    return v;
}
```

8、编写一个函数，函数原型为“int fun4(char\*a)”，请采用 while 循环求出由字符指针 a 所指

编写一个函数，函数原型为“int fun4(char\*a)”，请采用 while 循环求出由字符指针 a 所指向的字符串中包含的十进制数字的个数，并把把这个值作为函数值返回。

答案：

```
int fun4(char * a)
{
    int n=0;
    while(*a){
        if(*a>='0' && *a<='9') n++;
        a++;
    }
    return n;
}
```

9、编写一个主函数，计算  $1+3+3^2+\dots+3^{10}$  的值并输出，假定分别用 i,p,s 作为循环变量、累乘变量和累加变量的标识符。

程序为：

```
#include<stdio.h>
void main(){
    int i;
    int p=1;
    int s=1;
    for(i=1;i<=10;i++){p*=3; s+=p;}
    printf("%d\n",s); }
```

10、编写一个主函数，计算并输出  $1^2+2^2+\dots+n^2$  值，其中 n 值由键盘输入。

答：#include<stdio.h>

```
void main()
{
    int i; //用 i 作为计数（循环）变量
    int s=0; //用 s 作为累加变量
```

```
int n;
printf("输入一个自然数:");
scanf("%d",&n);
for(i=1; i<=n; i++) s+=i*i;
printf("s=%d\n",s);
}
```

11、编写一个主函数，假定 a 和 b 的取值范围为：

32. 编写一个主函数，假定 a 和 b 的取值范围为： $6 \leq a \leq 50, 10 \leq b \leq 30$ ，求出满足不定方程  $3a+2b=120$  的全部整数组解。如 (20,30) 就是其中的一组解。

答案：

```
.void main()
{
    int a,b; //1分
    for(a=6;a<=50;a++) //2分
        for(b=10;b<=30;b++) //3分
            if(3*a+2*b==120) printf("(%d, %d)\n",a,b); //6分
}
```

12、编写一个主函数，利用 while 循环，求出并显示满足不等式的最小 n 值。

答：#include<stdio.h>

```
void main()
{
    int i=0; double s=0; //或 int i=1; double s=1;
    while(s<5) s+=(double)1/++i;
    printf("n=%d\n",i);
}
```

13、编写一个主函数，利用 while 循环计算

31. 编写一个主函数，利用 while 循环计算  $1+2+4+\dots+2^{10}$  的值，最后输出计算结果。变量用 i 表示，每次累乘 2 的变量用 p 表示，每次累加一个数据项值的变量用 sum 表示。

答案：

```
.void main()
{
    int i=1, p=1, sum=1;
    while(i<=10) {p*=2; sum+=p; i++;}
    printf("sum=%d\n",sum);
}
```

14、编写一个主函数，求出满足不等式  $2^2+4^2+\dots+n^2<1000$  的最大 n 值，假定分别用 i 和 s 作为取偶数值和累加值的变量，并限定使用 do 循环编程。

答：#include<stdio.h>

```
void main()
{
    int i=0; //用 i 作为依次取值偶数的变量
    int s=0; //用 s 作为累加变量
    do {
        i+=2;
        s+=i*i;
    }while(s<1000);
    printf("n=%d\n",i-2);
}
```

15、补充完善下面的一个主函数，假定函数体中定义的整数变量 a 和 b 的取值范围分别为：

$6 \leq a \leq 50, 10 \leq b \leq 30$ ，求出满足不定方程  $3a+2b=120$  的全部整数组解。如 (20, 30) 就是其中的一组解。

void main()

```
{
    int a,b;
```

答案：

```
void main ()
```

```
{
    int a,b;
    for(a=6;a<=50;a++)//1分
        for(b=10;b<=30;b++)//3分
            if(3*a+2*b==120) printf("(%d, %d)\n",a,b); //6分
}
```

16、补充完整下面 xxk2 函数定义，假定函数参数表中 a 的值小于等于 b，函数功能是求出并返回 a 到 b 之间（包含 a 和 b 值在内）所有偶数之和。

```
int xxk2(int a, int b)
```

```
{
    int i, sum=0;
    //向下补充内容,用一个 for 循环实现

    return sum;
}
```

答案：

```
for(i=a; i<=b; i++)
```

```
    if(i%2==0) sum+=i;
```

17、补充完整下面的函数定义，该函数要求返回满足不等式

$1+2^2+3^2+\dots+n^2 \leq x$  最大的值。

```
int FF (int x) {
```

```
int n=1, s=1; //使用 n 作为计数变量，使用 s 作为累加和和
```

```
while (s<=x) {n++; s+=n*n;
```

```
//在下面一行写出合适的 while 循环语句
```

```
return n-1;
```

18、补充完整下面的函数定义，计算出  $1+1*3+2*3+ \dots +n*3$  的值并返回。

```
int FF(int n) {
```

```
    int i,s=1;
```

```
    //在下面添上一行 for 循环语句，循环变量为 i，累加变量为 s
```

```
    return s;
```

```
}
```

答案: for (i=1;i<=n;i++)s+=i \* 3; //4 分

19、补充完整下面的函数定义，计算出整型数组 a 中 n 个元素的所有偶数之和。

```
Int xxk4(inta[], int n) {
```

```
    Int i, s=0,
```

```
    // 在下面添上一行 for 循环语句，循环变量为 i，累加变量为 s
```

```
    Return s;
```

```
}
```

答案: for (i=0; i<n; i++) if (a[i]%2==0) s+=a[i]

20、补充完整下面的函数定义，利用 for 循环计算  $1+21+22$  十...+2“的值，最后返回计算结果。

补充完整下面的函数定义，利用 for 循环计算  $1+21+22$  十...+2“的值，最后返回计算结果。假定计数变量用 i 表示，每次累乘 2 的变量用 p 表示，每次累加一个数据项值的变量用 sum 表示。

```
int xxk8(int n)
```

```
{
```

```
int i, p=1,sum=1;
```

```
//补充完整下面 for 循环语句的循环体（即一条复合语句）
```

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

```
return sum;
```

```
}
```

答: {p\*=2;sum+=p;}

21、补充完整下面的函数定义，利用 for 循环计算  $1+4^1+4^2+\dots+4m$  的值，最后返回计算结果。

补充完整下面的函数定义，利用 for 循环计算  $1+4^1+4^2+\dots+4m$  的值，最后返回计算结果。假定计数变量用 i 表示，每次累乘 4 的变量用 p 表示，每次累加一个数据项值的变量用 sum 表示。

```
int xxk8 (int n)
```

```
{
```

```
int i, p=1, sum=1;
```

```
//补充完整下面 for 循环语句的循环体（即一条复合语句）
```

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

```
return sum;
```

```
}
```

```
{p*=4;sum+=p;}
```

22、补充完整下面的函数定义，要求返回二维数组 a[m][n]中所有元素的平均值。

补充完整下面的函数定义，要求返回二维数组 a[m][n]中所有元素的平均值。假定在计算过程中采用变量 s 存放累加值，采用 i 和 j 作为扫描数组的循环变量。

```
double Mean (doublea[][N], int m, int n)
```

```
{
```

```
int i, j;
```

```
double s=0.0;
```

```
//在下面添上一条合适的 for 双重循环语句
```

```
return s/(m *n);
```

```
}
```

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

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

```
s+=a[i][j];
```

23、补充完整下面的函数定义，要求返回二维数组 a[m][n]中所有元素的平均值。假定在计算过程中采用变量 s 存放累加值，采用 i 和 j 作为扫描数组的循环变量。

补充完整下面的函数定义，要求返回二维数组 a[m][n]中所有元素的平均值。假定在计算过程中采用变量 s 存放累加值，采用 i 和 j 作为扫描数组的循环变量。

```
double Mean(double a[][N],int m,int n)
```

```
{
```

```
Int i,j;
```

```
double s=0.0;
```

```
//在下面添上一条合适的 for 双重循环语句
```

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

```
for(j=0;j<n;j++)s+=a[i][j];
```

```
Return s/(m*n);
```

```
}
```

24、补充完整下面的函数定义，要求返回二维数组 a[m][n]中所有元素的平均值。假定在计算过程中采用变量 s 存放累加值，采用 i 和 j 作为扫描数组的循环变量。 double Mean (double a[][N], int m, int n) inti, j; doubles=0.0; /在下面添上合适的 for 双重循环语句 returns/ (m\*n) ;

评分标准:根据程序或函数编写的正确与完整程度酌情给分。

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

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

```
        s+=a[i][j];
```

25、补充完整下面函数 xxk5 的定义，计算并返回数组 a[n]中所有元素的平均值。

```
Double xxk5(int a[],int n)
```

```
{
```

```
    int i, sum=0, // 假定分别设置 i 和 sum 为循环变量和累加变量
```

```
    // 在下面添上一行 for 循环语句，循环变量为 i，累加变量为 sum
```

```
    Return double(sum)/n;
```

```
}
```

答案: for (i=0; i(n;i++) sum+=a[i];

26、补充完整下面函数定义中 while 循环的循环体，该函数的功能是求出并返回由字符指针 ptr 所指向的字符串中包含的小写英文字母的个数。

```
int fun(char * ptr)
```

```
{
```

```
    int n=0; //使用 n 作为统计变量
```

```
    while( * ptr) { //在下面补充合适的函数体内容
```

```
}
```

```
    return n;
```

```
}
```

```
if( * ptr>= 'a' && * ptr<= 'z') n++;
```

答: ptr++;

27、根据函数原型“double Mean(double a[M][N],int m ,int n)“，编写函数定义，要求返回二维数组 a[m][n]中所有元素的平均值。假定在计算过程中采用变量 v 存放累加值和最后的平均值。

```
double Mean(double a[M][N],int m ,int n) {
```

```
int i,j;
double v=0.0;
for(i=0; i<m; i++)
    for(j=0; j<n; j++) v+=a[i][j];
v/=m*n;
return v; } //注:函数体的最后两行可以合并为一条返回语句:
return v/=m*n;也可以把"="去掉。
```

28、根据函数原型“double Mean (double a[M][N],int m,int n)”，编写函数定义，要求返回二维

根据函数原型“double Mean (double a[M][N],int m,int n)”，编写函数定义，要求返回二维数组 a[m][n] 中所有元素的平均值。假定在计算过程中采用变量 S 存放累加值，采用 i 和 j 作为扫描数组的循环变量。

答案:

```
double Mean (double a[M][N], int m, int n)
```

```
{
int i,j;double s=0.0;
for (i=0;i<m;i++)
for (j=0;j<n;j++) s+=a[i][j];
return s/ (m*n);
}
```

29、根据函数原型“double Mean (double a[M][N],int m,int n)”，编写函数定义，要求返回二维数组 aCmlCal 中所有元素的平均值。假定在计算过程中采用变量 v 存放累加值和最后的平均值。

根据函数原型“double Mean (double a[M][N],int m,int n)”，编写函数定义，要求返回二维数组 aCmlCal 中所有元素的平均值。假定在计算过程中采用变量 v 存放累加值和最后的平均值。

答案:

```
double Mean (double a[M][N], int m, int n)
```

```
{
int i,j; double v=0.0;
for (i=0; i<m; i++)
    for (j=0; j<n; j++) v+=a[i][j];
v/=m*n;
return v;
}
```

30、根据函数原型“int FF(int a[],int n)”，编写函数定义，计算并返回数组 a[n] 中所有元素之和。

假定分别设置 i 和 sum 为循环变量和累加变量。

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

```
{
    int i,sum=0;
//请在下面添加若干语句
}
```

答案: for (i=0; i<n; i++)sum+=a[i]; //2 分

```
return sum; //4 分
```

31、根据函数原型“int FF(int a[],int n)”，编写函数定义，计算并返回数组 a[n] 中所有元素之和。

程序为:

```
int FF(int a[], int n) {
    int i,sum=0;
```

```
for(i=0; i<n; i++) sum+=a[i];
return sum; }
```

32、根据函数原型“int MM(int a[], int m)”，编写函数定义，计算并返回数组 a[m] 中元素最大值和最小值之差。

```
int MM(int a[], int m){
int i,x1,x2;
x1=x2=a[0];
for(i=1; i<m; i++){
if(a[i]>x1) x1=a[i];
if(a[i]<x2) x2=a[i];
}
return x1-x2;}
```

33、根据函数原型“int MM(int a[], int m)”，编写函数定义，计算并返回数组 a[m] 中所有元素

根据函数原型“int MM(int a[], int m)”，编写函数定义，计算并返回数组 a[m] 中所有元素的最大值的平方值，假定用变量 max 保存最大值。

答案:

```
int MM(int a[],int m){
int i,max=a[0];
for(i=1;i<m;i++)
if(a[i]>max)max=a[i]
return max * max;
}
```

34、根据函数原型“int DD (inta[], intn,intMM)”，编写函数定义，利用双重 for 循环统计出数组 a[n]

根据函数原型“int DD (inta[], intn,intMM)”，编写函数定义，利用双重 for 循环统计出数组 a[n] 中任何两个不同元素的值等于 MM 值的配对个数，然后返回统计结果。如假定 a[i]+a[j] 等于 MM，同时 i≠j，则为一个配对。

```
intDD (inta[], intn,intMM)
```

```
{
inti,j,C=0; //用 i 和 j 作为循环变量，用 C 作为统计变量
//在下面补充一个双重 for 循环语句
Return C;
}
```

答案: for (i=0; i<n; i++) //1 分

```
for (j=i+1; j<n; j++) //2 分
```

```
if (a[i]+a[j]==MM) C++; //6 分
```

35、根据函数原型“int FF (inta[], int n)”，编写函数定义，计算并返回数组 a[n] 中所有元素之和。

```
答案: int FF(int a[],int n){
```

```
int i, sum=0;
for(i=0;i<n;i++)sum+=a[i];
return sum;
```

36、根据函数原型“int MM (intaE[],intrn)”，编写函数定义，计算并返回数组 aEm) 中元素最大值和最小值之差。

根据函数原型“int MM (intaE[],intrn)”，编写函数定义，计算并返回数组 aEm) 中元素最大值和最小值之差。

答案:

```
int MM(int a[],int m){
```

```
int i,x1,x2; x1=x2=a[0];
for(i=1; i<m; i++){
if(a[i]>x1) x1=a[i];
if(a[i]<x2) x2=a[i];
}
return x1-x2;
}
```

37、根据函数原型“void DD (int a[], int n, int MM)”，编写函数定义，利用双重 for 循环统计出数组 a[n] 中任何两个不同元素的值等于 MM 值的配对个数，然后返回统计结果。如假定 a[i]+a[j] 等于 MM，同时 i≠j，则为一个配对。

答案:

```
int DD(int a[],int n,int MM){
int i,j,c=0;
for (i=0; i<n; i++)
for (j=i+1;j<n;j++)
if (a[i]+a[j]==MM) c++;
return c;
}
```

38、完善下面程序，该程序功能是从键盘上

31. 完善下面程序,该程序功能是从键盘上输入 10 个整数到一维数组,下标从大到小的次序输出数组 a[10]中的每个元素值,元素值之间用一个空

```
#include<stdio.h>

void main()
{
    int a[10],i; //用 i 作为循环变量
    printf("从键盘输入 10 个整数: \n");
    //向下补充程序内容,用两个 for 循环实现
    printf("\n");
}
```

答案: ↓

//向下补充程序内容,用两个 for 循环实现

```
for(i=0; i<10; i++) scanf("%d",&a[i]); //3 分
for(i=9; i>=0; i--) printf("%d ",a[i]); //6 分
```

注:第 1 行中的 &a[i]也可替换为 a+i。

31. 完善下面程序,该程序功能是从键盘上输入 10 个整数到一维数组,下标从大到小的次序输出数组 a[10]中的每个元素值,元素值之间用一个

```
#include<stdio.h>

void main()
{
    int a[10],i; //用 i 作为循环变量
    printf("从键盘输入 10 个整数: \n");
    //向下补充程序内容,用两个 for 循环实现
    printf("\n");
}
```

答案: ↓

//向下补充程序内容,用两个 for 循环实现

```
for(i=0; i<10; i++) scanf("%d",&a[i]); //3 分
for(i=9; i>=0; i--) printf("%d ",a[i]); //6 分
```

注:第 1 行中的 &a[i]也可替换为 a+i。

39. 完善下面程序中 xxkl 函数的定义,使程序能够输出数组 a 中所有元素的最大值。

```
#include<stdio.h>

int xxkl(int a[], int n);

void main()
{
    int x;
    int a[6]={15,20,36,48,24,44};
    x=xxkl(a,6);
    printf("x=%d\n", x);
}
```

int xxkl(int a[], int n)

```
{
    int i, x=a[0], max=a[0];
    //向下补充内容,用一个 for 循环实现
}
```

答案:

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

```
if(a[i]>max) max=a[i];
```

程序填空题(178)--电大资源网: <http://www.dda123.cn/> (微信搜: 905080280)

- 1、 #include<stdio.h> Void main(){ intx=...
- 2、 #include void main () { int a[8]={3,5,7,9,...
- 3、 #include int Count(int a[], int n, int x,...
- 4、 #include int LA(int \*a, int n, int x) {
- 5、 #include int WF(int x, int y) {
- 6、 #include struct Worker {
- 7、 #include struct Worker {
- 8、 #include struct Worker {char name[15]; i,...
- 9、 #include<stdio.h> int LA(int \*a, i,...
- 10、 #include<stdio.h>
- 11、 #include<stdio.h> int LA(int \*a, int n) {...
- 12、 #include<stdio.h> int WF(int x, int y)...
- 13、 #include<stdio.h> void main () { int a[ 3 ...
- 14、 #include<stdio.h> void main () { int a[8]=...
- 15、 #include<stdio.h> void main () { int i,j,...
- 16、 #include<stdio.h> void main () { int x=20;...
- 17、 #include<stdio.h> # define N 6
- 18、 #include<stdio.h> # define M 6
- 19、 #include<stdio.h> # define M 6
- 20、 #include<stdio.h> # define N 6

- 21、 #include<stdio.h> #define M 6
- 22、 #include<stdio.h> #define N 6
- 23、 #include<stdio.h> #define N6
- 24、 #include<stdio.h> #include<string.h>...
- 25、 #include<stdio.h> #include<string.h>...
- 26、 #include<stdio.h> #include<string.h> str...
- 27、 #include<stdio.h> #include<string.h> voi...
- 28、 #include<stdio.h> #include<string.h> voi...
- 29、 #include<stdio.h> const int M=20;
- 30、 #include<stdio.h> const int M=20;
- 31、 #include<stdio.h> const int M=20;
- 32、 #include<stdio.h> int a[10]={4,5,6,15,20...
- 33、 #include<stdio.h> int a[6]={4,5,6,15,20...
- 34、 #include<stdio.h> int a[6]={4,5,6,15,20...
- 35、 #include<stdio.h> int Count(int a[], int,...
- 36、 #include<stdio.h> int LA(int \*a, int n)...
- 37、 #include<stdio.h> int LA(int \*a, int n) ...
- 38、 #include<stdio.h> int LA(int \*a, int n, ...
- 39、 #include<stdio.h> int LB(int \*a, int n)...
- 40、 #include<stdio.h> int LB(int \*a, int n) ...
- 41、 #include<stdio.h> int WF(int x, int y) {...
- 42、 #include<stdio.h> int WF(int x, int y) {...
- 43、 #include<stdio.h> int WF(int x, int y) {...
- 44、 #include<stdio.h> int WF(int x, int y) {...
- 45、 #include<stdio.h> struct Worker (char na...
- 46、 #include<stdio.h> struct Worker (char na...
- 47、 #include<stdio.h> struct Worker {
- 48、 #include<stdio.h> struct Worker {
- 49、 #include<stdio.h> struct Worker { char n...
- 50、 #include<stdio.h> struct Worker {char na...
- 51、 #include<stdio.h> struct Workerichar nam...
- 52、 #include<stdio.h> struer Worker {char na...
- 53、 #include<stdio.h> void LE(int \*a, int \*...
- 54、 #include<stdio.h> void LE(int\* a, int\* b)...
- 55、 #include<stdio.h> void main () { int n=4,v...
- 56、 #include<stdio.h> void main () {char s[]=\...
- 57、 #include<stdio.h> void main ()
- 58、 #include<stdio.h> void main ()
- 59、 #include<stdio.h> void main () {...
- 60、 #include<stdio.h> void main () {...
- 61、 #include<stdio.h> void main () {...
- 62、 #include<stdio.h> void main () {...
- 63、 #include<stdio.h> void main () {...
- 64、 #include<stdio.h> void main () {...
- 65、 #include<stdio.h> void main () {...
- 66、 #include<stdio.h> void main () { char a []=...
- 67、 #include<stdio.h> void main () { char a[] =...
- 68、 #include<stdio.h> void main () { char a[] =...
- 69、 #include<stdio.h> void main () { int a,b;...
- 70、 #include<stdio.h> void main () { int a[10]=...
- 71、 #include<stdio.h> void main () { int a[3][3]...

72. #include<stdio.h> void main () { int a[6]= ...  
 73. #include<stdio.h> void main () { int a[8]= ...  
 74. #include<stdio.h> void main () { int a[8]= ...  
 75. #include<stdio.h> void main () { int a[8]=(...  
 76. #include<stdio.h> void main () { int a[8]=(...  
 77. #include<stdio.h> void main () { int a[8]=1...  
 78. #include<stdio.h> void main () { int a[8]=...  
 79. #include<stdio.h> void main () { int a[9]= ...  
 80. #include<stdio.h> void main () { int a[9]=...  
 81. #include<stdio.h> void main () { int a, b;...  
 82. #include<stdio.h> void main () { int f1,f2,...  
 83. #include<stdio.h> void main () { int i, s1=...  
 84. #include<stdio.h> void main () { int i, s=0...  
 85. #include<stdio.h> void main () { int i, s=0...  
 86. #include<stdio.h> void main () { int i,j,k=...  
 87. #include<stdio.h> void main () { int i,j,k=...  
 88. #include<stdio.h> void main () { int i,s=0;...  
 89. #include<stdio.h> void main () { int i,s=0;...  
 90. #include<stdio.h> void main () { int i,s=0;...  
 91. #include<stdio.h> void main () { int i;...  
 92. #include<stdio.h> void main () { int i, s=0;...  
 93. #include<stdio.h> void main () { int n=4, v...  
 94. #include<stdio.h> void main () { int n=6, v...  
 95. #include<stdio.h> void main () { int x=20...  
 96. #include<stdio.h> void main () { int x=20...  
 97. #include<stdio.h> void main () { int x=5;...  
 98. #include<stdio.h> void main () { struct Wor...  
 99. #include<stdio.h> void main () {int i, s=0;...  
 100. #include<stdio.h> void main () {int i,j,k=0...  
 101. #include<stdio.h> void main () {int i,s=0;...  
 102. #include<stdio.h> void main () { int a[3][3]...  
 103. #include<stdio.h> void SB(char ch) {...  
 104. #include<stdio.h> void SB(char ch) { swi...  
 105. #include<stdio.h> void SB(char ch)  
 106. #include<stdio.h> Voidmain () 。 Intx=5,v=15...  
 107. #include<stdio.h> void main () 。 int a[3]...  
 108. #include<stdio.h> void main () 。 int i=0...  
 109. #include<stdio.h> void main(){ int a[3]...  
 110. #include<stdio.h> void main(){int a[8]=...  
 111. #include<stdio.h> # define N 8 void m...  
 112. #include<stdio.h> void main () { int a...  
 113. #include<stdio.h> void main () { int x...  
 114. #include<stdio.h> void main () {int a[...  
 115. #include<stdio.h> void main () {int a[...  
 116. #include<stdio.h> void main () {int a[...  
 117. #include<stdio.h> Void main() { Int a...  
 118. #include<stdio.h> void main() {int i=1...  
 119. #include<stdio.h> Void main(){ inti, ...  
 120. #include<stdio.h> void main () {int i=0 ...  
 121. #include<stdio.h> void main(){ int a[3]...  
 122. #include<stdio.h> void main () { int x=5 ...

123. #include<stdio.h> void main () { int x=5,v...  
 124. #include #define M6  
 125. #include #define NS  
 126. #include<stdio.h> #defineN5 void main () {...  
 127. #include<stdio.h> #void main () {inti, fl=1...  
 128. #include<stdio.h> struct Worker{char na...  
 129. #include<stdio.h> void main () { inta~25, b=...  
 130. #include<Stdio.h> void main () { inta[8]=...  
 131. #include<stdio.h> void main () { inta (8) ...  
 132. #include<stdio.h> void main () { intn=0, v=0;...  
 133. #include<stdio.h> void main () {inta (8) =...  
 134. #include # define N 8  
 135. #include<stdio.h> #define N 5 void main () ...  
 136. #include<stdio.h> void main () { int x=12...  
 137. #include<stdio.h> void main(){inti=0, s=...  
 138. #include<stdio.h> void main () { int a[8]=...  
 139. #include<stdio.h> void main () { int a[8] =...  
 140. #include<stdio.h> void main () { int i,f1=1...  
 141. #include<stdio.h> void main () { int x=5 ,v...  
 142. #include<stdio.h> void main () { int x=5,v=...  
 143. #include<stdio.h> void main () { int x=5,v=...  
 144. #include<stdio.h> void main () { inti, s1=0...  
 145. #include<stdio.h> void main () { intx=1,v; ...  
 146. #include<stdio.h> void main () { int a=5, b=8, c...  
 147. #include<stdio.h> void main () { int a[4][3]...  
 148. #include<stdio.h> void main () { int x=5, y=8, z...  
 149. #include<stdio.h> void main () { inta[6] =...  
 150. #include<stdio.h> void main () { inti, s=0;...  
 151. #include void SB(char ch){  
 152. #include<stdio.h> voidmain () { int n=2, y=0; wh...  
 153. #include void main () 。  
 154. #include<stdio.h> void main(  
 155. #include<stdio.h> void main () { int x=5...  
 156. #include<stdio.h> void main () {int fl=1,f2...  
 157. #include<stdio.h>#defineN6  
 158. #include<stdio.h>intWF(intx,inty){  
 159. #include<stdio.h>void main () { int a,bv...  
 160. #include<stdio.h>void main () { int i;...  
 161. #include<stdio.h>void main () { int i,j,k=0;...  
 162. #include<stdio.h>voidLE(inta,intb){  
 163. #include<stdio.h>voidmain () {int i,s=0;...  
 164. #include<stdio.h>[intWF(intx,inty)]...  
 165. #includeconstintM=20  
 166. #includeinta[10]={4,5,6,15,20,13,12,7,8,9...  
 167. #includevoidfun(charss[])  
 168. #includevoidmain ()  
 169. #includevoidmain ()  
 170. #includevoidmain () {  
 171. #includevoidmain () {  
 172. #includevoidmain () {  
 173. #includevoidmain () {

174. #includevoidmain () {  
 175. void main () { int a[3][3] =1(3,5,7),(9,11...  
 176. 按照“int FF (inta[], intn)”的函数声明, 编写出...  
 177. 当使用 xkZC 调用此函数时, 返回值为() ...  
 178. 假定使用 xxk3 (4) 调用下面函数, 则返回值为 () ...  
 1. #include Void main(){ intx=15, y=8,z=4;  
 # include<stdio.h>  
 Void main(){  
 intx=15, y=8,z=4;  
 if(x>y)x--y;  
 if (x>z)x-z ;  
 printf("%d\n", x);  
 }  
 答案: 4  
 2. #include void main () { int a[8]={3,5,7,9,2,3,4,8};  
 #include  
 void main() {  
 int a[8]={3,5,7,9,2,3,4,8};  
 int s=0,\*p;  
 for(p=a;p<a+8;) s+=\*p++;  
 printf("s=%d\n",s);  
 }  
 答案: s=41  
 3. #include int Count(int a[], int n, int x)  
 #include  
 int Count(int a[], int n, int x)  
 {  
 int i,c=0;  
 for(i=0;i<n;i++) if(a>x) c++;  
 return c;  
 }  
 void main() {  
 int a[8]={20,15,32,47,24,36,28,70};  
 int b=Count(a,8,30);  
 printf("b=%d\n",b);  
 }  
 答案: B=4  
 4. #include int LA(int \*a, int n, int x) {  
 #include  
 int LA(int \*a, int n, int x) {  
 int i,s=0;  
 for(i=0;i<n;i++)  
 if(a<x) s+=a;  
 return s;  
 }  
 void main() {  
 int a[8]={5,10,15,8,12,3,9,20};  
 int b=LA(a,5,10);  
 int c=LA(a+2,6,10);  
 printf("%d %d\n",b,c);  
 }

答案： 13 20

5、**#include int WF(int x, int y) {**

```
#include
int WF(int x, int y) {
x=x+y;
y=x+y;
return x+y;
}
void main() {
int x=5, y=7;
int z=WF(x,y);
printf("z=%d\n",z);
}
```

答案： z=31

6、**#include struct Worker {**

```
#include
struct Worker {
char name[15]; //姓名
int age; //年龄
float pay; //工资
};
void main() {
struct Worker x={ wanghua,52,4300};
struct Worker y, *p;
y=x; p=&x;
printf("%s %d %6.2f\n,x.name,y.age,p-&gt;pay);
}
```

答案： wanghua 52 4300

7、**#include struct Worker {**

```
#include
struct Worker {
char name[15]; //姓名
int age; //年龄
float pay; //工资
};
void main() {
struct Worker x;
char *t=liouting;
int d=38; float f=3493;
strcpy(x.name,t);
x.age=d; x.pay=f;
printf("%s %d %6.0f\n,x.name,x.age,x.pay);
}
```

答案： liouting 39 2493

8、**#include struct Worker {char name[15]; int age; float pay;};**

```
#include
struct Worker {char name[15]; int age; float pay;};
int Less(struct Worker r1, struct Worker r2) {
```

```
if(r1.age<r2.age) return 1; else return 0;
}
void main() {
struct Worker a[4]={ {abc,25,2420},{def,58,4638},
{ghi,49,4260},{jkl,36,3750}};
struct Worker x=a[0];
int i;
for(i=1; i<4; i++)
if(Less(x,a)) x=a;
printf("%s %d %6.0f\n,x.name,x.age,x.pay);
}
```

答案： def 58 4638

9、**#include<stdio. h> int LA(int \* a, int n){**

```
#include<stdio. h>
int LA(int * a, int n){
int i, s=0;
for(i=0;i<n; i++)s+=a[i]
return s;
}
void main(){
int a[5] ={1, 2, 3, 4, 5};
int b=LA(a,5)+LA(a+1, 3)
printf"b=%An", b);
}
```

答案： b=24

10、**#include<stdio. h>**

```
#include<stdio. h>
void main(){
int x=20;
int i=2;
while(i<x){
if(x%i==0){printf"%d", i);x/=i;}
i++;
}
}
```

答案： 2 5

11、**#include<stdio. h> int LA(int \*a, int n) {**

```
#include<stdio. h>
int LA(int *a, int n) {
int i,s=0;
for(i=0; i<n;i++) s+=a[i];
return s; }
void main() {
int a[5]={1 ,2,3,4,5};
int b=LA(a,5)+LA(a+2, 3);
printf("b=%d\n",b);}
运行结果： b=27
```

12、**#include<stdio. h> int WF(int x , int y) {**

```
#include<stdio. h>
int WF(int x , int y) {
x=x+y;
```

```
y+=x;
return x+y; }
void main() {
int x=3 , y=8;
printf("%d\n" ,WF(x,y));
```

运行结果： 30

13、**#include<stdio. h> void main ( ) { int a[ 3 J [ 31 = { { 3, 8, 6 } ,**

{ 5, 9, 2 } , { 8, 5, 12 } } ;

答： s=30

14、**#include<stdio. h> void main ( ) { int**

**a[8]={76,63,54,95,40,75,90,66};**

**#include<stdio. h>**

```
void main() {
int a[8]={76,63,54,95,40,75,90,66};
int i, s=0;
for(i=0;i<8;i++)
if(a[i]>=70 && a[i]<=90) s+=a[i];
printf("s=%d\n",s) ; }
```

运行结果： s=241

15、**#include<stdio. h> void main ( ) { int i,j ,k==0;**

**#include<stdio. h>**

```
void main() {
int i,j ,k==0;
for ({i=0;i<5;i++)
for(j=i;j<5;j++) k++;;
printf("%d\n",k) ; }
```

运行结果： 15

16、**#include<stdio. h> void main ( ) { int x=20;**

**#include<stdio. h>**

```
void main() {
int x=20;
int i=2;
while(i<x) {
if(x%i==0) {printf("%d",i); x/=i;}
i++; } }
```

运行结果： 25

17、**#include<stdio. h># define N 6**

**}**

执行下面程序的输出结果为（13）。

18、**#include<stdio.h> # define M 6**

```
27. #include<stdio.h>
#define M 6
void main()
{
    int i,x1=0,x2=0;
    int a[M]={10,15,22,24,12,30};
    for(i=0; i<M/2; i++) {x1+=a[i]; x2+=a[M-i-1];}
    printf("%d%d\n",x1,x2);
}
```

输出结果:

答案: 47 66

19、#include<stdio.h> #define M 6

```
27. #include<stdio.h>
#define M 6
void main()
{
    int i,x1=0,x2=0;
    int a[M]={2,5,8,4,6,9};
    for(i=0; i<M/2; i++) {x1+=a[i]; x2+=a[M-i-1];}
    printf("%d%d\n",x1,x2);
}
```

输出结果:

答案: 15 19

20、#include<stdio.h> #define N 6

```
#include<stdio.h>
#define N 6
void main()
{
    int i,a[N]={2,5,8,10,15,21};
    for(i=0; i<N; i++)
        if(a[i]%5==0) printf("%d ",a[i]);
    printf("\n");
}
```

答案: 5 10 15

21、#include<stdio.h> #define M 6

```
#include<stdio.h>
#define M 6
void main() {
    int i,x;
    int a[M]={10,15,22,37,46,58};
    for(i=0; i<M/2; i++)
        {x=a[i]; a[i]=a[M-1-i]; a[M-1-i]=x;}
    for(i=0; i<6; i++) printf("%d ",a[i]);
    printf("\n"); }
```

运行结果: 58 46 37 22 15 10

22、#include<stdio.h> #define N 6

```
#include<stdio.h>
#define N 6
void main() {
    int i, a[N]={2, 5, 8, 10, 15, 21};
    for(i=0; i<N; i++)
        if(a[i]%5) printf("%d ", a[i]);
    printf("\n"); }
```

运行结果: 2 8 21

23、#include<stdio.h> #define N6

```
执行下面程序的输出结果为 ( )。
#include<stdio.h>
#define N6
void main ( )。
int a[N];
int i;
a[0]=1; a[1]=2;
for (i=2; i<N; i++) a[i]=a[i-2]+a[i-1]+1;
printf ( " %d\n " ,a[N-1] )。
)
```

答: 20

24、#include<stdio.h> #include<string.h>

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

```
#include<string.h>
void fun(char ss[]);
void main( ) {
    char s[15]="0123456789";
    fun(s);
    printf("%s\n",s);
}
void fun(char ss[]) {
    int i, n=strlen(ss);
    for(i=0; i<n/2; i++) {
        char c=ss[i];
        ss[i]=ss[n-1-i];
        ss[n-1-i]=c;
    }
}
```

答案: 9876543210

25、#include<stdio.h> #include<string.h>

```
#include<stdio.h>
#include<string.h>
struct Worker {
    char name[15]; //姓名
    int age; //年龄
    float pay; //工资
};
void main() {
    struct Worker x;
    char *t="liouting";
    int d=38; float f=3493;
    strcpy(x.name,t);
    x.age=d; x.pay=f;
    printf("%s %d %6.0f\n",x.name,x.age,x.pay);
}
```

答案: liouting 39 2493

26、#include<stdio.h> #include<string.h> struct Worker { char name[15]; int age; float pay;};

```
#include<stdio.h>
#include<string.h>
struct Worker { char name[15]; int age; float pay;};
void main() {
    struct Worker x;
    char *t="liouting";
    int d=38; float f=400;
    strcpy(x.name,t);
    x.age=d; x.pay=f;
    x.age++; x.pay*=2;
    printf("%s %d %6.2f\n",x.name,x.age,x.pay); }
```

运行结果: liouting 39 800.00

27、#include<stdio.h> #include<string.h> void main() {

```
#include<stdio.h>
#include<string.h>
void main() {
```

```

char s[15]="567891234";
int i, n=strlen(s);
for(i=0; i<n/2; i++) {
    char c=s[i];
    s[i]=s[n-1-i];
    s[n-1-i]=c; }
printf("%s\n",s); }
运行结果: 432198765
28、#include<stdio.h> #include<string.h> void main () { int i;
#include<stdio.h>
#include<string.h>
void main() {
    int i;
    unsigned int len;
    char* a[5]={"student", "worker", "cadre", "soldier",
"zzeasan123"};
    len=strlen(a[0]);
    for(i=1; i<5; i++)
        if(strlen(a[i])>len) len=strlen(a[i]);
    printf("%d\n", len); }
运行结果: 10
29、#include<stdio.h> const int M=20;
答案: k=4
30、#include<stdio.h> const int M=20;
#include<stdio.h>
const int M=20;
void main() {
    int i=2;
    while(1) {
        if(i>M/2) eak;
        if(M%i==0) printf("%d ",i);
        i++; }
    printf("\n"); }
运行结果: 2 4 5 10
31、#include<stdio.h> const int M=20;
#include<stdio.h>
const int M=20;
void main()
{
    int i,c2,c3,c5;
    c2=c3=c5=0;
    for(i=1; i<=M; i++) {
        if(i%2==0) c2++;
        if(i%3==0) c3++;
        if(i%5==0) c5++;
    }
    printf("%d %d %d\n",c2,c3,c5);
}
答案: ①10 6 4
32、#include<stdio.h> int a[10]={4,5,6,15,20,13,12,7,8,9};
#include<stdio.h>

```

```

int a[10]={4,5,6,15,20,13,12,7,8,9};
void main()
{
    int i,s[4]=0;
    for(i=0; i<10; i++) {
        switch(a[i]%4) {
            case 0: s[0]+=a[i];eak;
            case 1: s[1]+=a[i];eak;
            case 2: s[2]+=a[i];eak;
            default: s[3]+=a[i];eak;
        }
    }
    for(i=0; i<4; i++) printf("%d ",s[i]);
}
答案: ①44 27 6 22
33、#include<stdio.h> int a[6]= {4,5,6,15,20,12};
答案: 29 33
34、#include<stdio.h> int a[6]={4,5,6,15,20,12};
#include<stdio.h>
int a[6]={4,5,6,15,20,12};
void main() {
    int i,s1,s2;
    s1=s2=0;
    for(i=0; i<6; i++) {
        switch(a[i]%2) {
            case 0: s2+=a[i];eak;
            case 1: s1+=a[i];eak; } }
    printf("%d %d\n",s1,s2); }
运行结果: 20 42
35、#include<stdio.h> int Count(int a[], int n, int x)
#include<stdio.h>
int Count(int a[], int n, int x)
{
    int i,c=0;
    for(i=0;i<n;i++)
        if(a[i]>x) c++;
    return c;
}
void main() {
    int a[8]={20,15,32,47,24,36,28,70};
    int b=Count(a,8,30);
    printf("b=%d\n",b);
}
答案: B=4
36、#include<stdio.h> int LA(int * a, int n) {
答案: b=27
37、#include<stdio.h> int LA(int *a, int n) {
#include<stdio.h>
int LA(int *a, int n) {
    int i,s=0;
    for(i=0;i<n;i++) s+=a[i];
}

```

```

return s; }
void main() {
    int a[5]={1,2,3,4,5};
    int b=LA(a,5)+LA(a+1,3);
    printf("b=%d\n",b); }
运行结果: b=24
38、#include<stdio.h> int LA(int *a, int n, int x) {
#include<stdio.h>
int LA(int *a, int n, int x) {
    int i,s=0;
    for(i=0;i<n;i++)
        if(a[i]<x) s+=a[i];
    return s;
}
void main() {
    int a[8]={5,10,15,8,12,3,9,20};
    int b=LA(a,5,10);
    int c=LA(a+2,6,10);
    printf("%d %d\n",b,c);
}
答案: 13 20
39、#include<stdio.h> int LB(int * a, int n)

28. #include<stdio.h>
int LB(int * a, int n) {
    int * p,s=1;
    for(p=a;p<a+n;p++) s *= * p;
    return s;
}
void main() {
    int a[]={1,2,3,4,5};
    int b=LB(a,5);
    printf("b=%d\n",b);
}
输出结果:
答案: b=120
40、#include<stdio.h> int LB(int *a, int n) {
#include<stdio.h>
int LB(int *a, int n) {
    int i,s=1;
    for(i=0;i<n;i++) s*=*a++;
    return s; }

```

```
void main() {
    int a[]={1,2,3,4,2,4,5,2};
    int b=LB(a,4)+LB(a+3,3);
    printf("b=%d\n",b); }
运行结果: b=56
```

41、#include<stdio.h> int WF(int x, in: y) {

```
27. #include<stdio. h>
    int WF(int x, in: y) {
        x= x+y;
        y+= x;
        return x+y;
    }
void main() {
    int x=3, y=5;
    printf("%d\n", WF(x,y));
}
```

答案: 21

42、#include<stdio.h> int WF(int x, int y) {

```
4. #include<stdio. h>
    int WF(int x, int y) {
        x= x+y;
        y+= x;
        return x+y;
    }
void main() {
    int x=3, y=8;
    printf("%d\n", WF(x,y));
}
```

答案: 30

43、#include<stdio.h> int WF(int x, int y) {  
#include<stdio.h>

```
int WF(int x, int y) {
    x=x+y;
    y=x+y;
    return x+y;
}
void main() {
```

```
int x=5, y=7;
int z=WF(x,y);
printf("z=%d\n",z);
}
```

答案: z=31

44、#include<stdio.h> int WF(int x, int y) { x=x+y;

```
#include<stdio.h>
int WF(int x, int y) {
    x=x+y;
    y+=x;
    return x+y; }
void main() {
    int x=3, y=5;
    printf("%d\n",WF(x,y)); }
```

运行结果: 21

45、#include<stdio.h> struct Worker (char name[15]; int age; double pay);

```
28. #include<stdio. h>
    struct Worker {char name[15]; int age;double pay;};
    void main(){
        struct Worker a[4]={{ "abc",20,2400},{ "defa",50,3600}
            {"ghin",40,1800},{ "jkl",36,2000}};
        int i;double s=0;
        for(i=0;i<4;i++)s+=a[i]. pay;
        printf("s= %5. 0Lf\n",s);
    }
```

输出结果:

答案: s=9800

46、#include<stdio.h> struct Worker (char name[15]; int age; float pay):

```
28. #include<stdio. h>
    struct Worker {char name[15]; int age; float pay;};
    void main() {
        struct Worker x={"wanghua",48,2500};
        struct Worker y, * p;
        y=x; p=&x;
        printf("%d %7. 2f\n", y. age+p->age, p->pay+60);
    }
```

输出结果:

答案: 96 2560.00

47、#include<stdio.h> struct Worker {

```
#include<stdio.h>
struct Worker {
    char name[15]; int age; float pay; };
void main() {
    struct Worker x={"wanghua",52,2350};
    struct Worker y, *p;
    y=x; p=&x;
    printf("%d %7.2f\n", y.age+p->age, p->pay+20); }
```

运行结果: 104 2370.00

48、#include<stdio.h> struct Worker {

```
#include<stdio.h>
struct Worker {
    char name[15]; //姓名
    int age; //年龄
    float pay; //工资
};
void main() {
    struct Worker x={"wanghua",52,4300};
    struct Worker y, *p;
    y=x; p=&x;
    printf("%s %d %6.2f\n",x.name,y.age,p->pay);
}
```

答案: wanghua 52 4300

49、#include<stdio.h> struct Worker { char name[15]; int age; double pay);

```
5. #include<stdio. h>
    struct Worker { char name[15]; int age; double pay;};
    void main() {
        struct Worker a[4]={{ "abc",20,2420},{ "defa",50,3638},
            {"ghin",40,2560},{ "jkl",36,2375}};
        int i, s=0;
        for(i=0; i<4; i++) s+=a[i]. age;
        printf("s= %5. 2Lf\n", (double)s/4);
    }
```

答案: s=36.50

50、#include<stdio.h> struct Worker {char name[15]; int age; float pay;};

```
#include<stdio.h>
struct Worker {char name[15]; int age; float pay;};
int Less(struct Worker r1, struct Worker r2) {
    if(r1.age<r2.age) return 1; else return 0;
}
void main() {
    struct Worker a[4]={{ "abc",25,2420},{ "def",58,4638},
```

```

    {"ghi",49,4260},{ "jkl",36,3750}};
    struct Worker x=a[0];
    int i;
    for(i=1; i<4; i++)
        if(Less(x,a[i])) x=a[i];
    printf("%s %d %6.0f\n",x.name,x.age,x.pay);
}

```

答案: def 58 4638

51、#include<stdio.h> struct Worker{char name[15]; int age; float pay;

```

28. #include<stdio.h>

    struct Worker{char name[15];int age;float pay;};

    void main(){

        struct Worker x={"wanghua",48,2500};

        struct Worker y,*p;

        y=x;p=&x;

        printf("%d\n",y.age+p->age);

    }

```

输出结果: 96

答案: 96

52、#include<stdio.h> struer Worker {char name[15]; int age; float pay;}

```

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

    struct Worker {char name[15]; int age; float pay;};

    void main() {

        struct Worker x;

        char *t="liouting";

        int d=38; float f=400;

        strcpy(x.name,t);

        x.age=d;x.pay=f;

        x.age--;x.pay/=2;

        printf("%s%d%6.2f\n",x.name,x.age,x.pay);

    }

```

输出结果:

答案: liouting 37 200.00

53、#include<stdio.h> void LE(int \* a, int \* b) {

```

#include<stdio.h>

    void LE(int * a, int * b) {

        int x=*a;

        *a=*b; *b=x;

    }

    void main() {

        int x=15, y=26;

        LE(&x,&y);

        printf("%d %d\n",x,y);

    }

```

答案: 26 15

54、#include<stdio.h> void LE(int\* a, int\* b) {

```

#include<stdio.h>

    void LE(int* a, int* b) {

        int x=*a;

        *a=*b; *b=x; }

    void main() {

        int x=15, y=26;

        printf("%d %d\n", x, y);

        LE(&x, &y);

        printf("%d %d\n", x, y); }

```

运行结果: 15 26 26 15

55、#include<stdio.h> void main () { int n=4,y=0;

```

24. #include<stdio.h>

    void main(){

        int n=4,y=0;

        while(n-->0)y+=3;

        printf("y=%d\n",y);

    }

```

输出结果:

答案: y=12

56、#include<stdio.h> void main () {char s[]="123456789\1

```

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

    void main() {

        char s[]="123456789";

        int i, n=strlen(s);

        for(i=0; i<n/2; i++) {

            char c=s[i]; s[i]=s[n-1-i]; s[n-1-i]=c;

        }

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

    }

```

输出结果:

答案: 987654321

57、#include<stdio.h> void main ()

```

#include<stdio.h>

    void main()

    {

        int x=5;

        switch(2*x-3) {

            case 4: printf("%d ",x);

            case 7: printf("%d ",2*x+1);

            case 10: printf("%d ",3*x-1); eak;

            default: printf("%s","default\n");

        }

    }

```

答案: 11 14

58、#include<stdio.h> void main ()

```

#include<stdio.h>

    void main()

    {

        int i,s;

        for(i=1,s=0;i<10;i++) {

            if(i%3==0) continue;

            s+=i;

        }

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

    }

```

答案: ①27

59、#include<stdio.h> void main () {

```

#include<stdio.h>

    void main() {

```

```
int i, s1=0, s2=0;
for(i=0;i<10;i++)
if(i%2) s1+=i; else s2+=i;
printf("%d %d\n",s1,s2);
}
```

答案: 25 20

```
60、#include<stdio.h> void main () {
#include<stdio.h>
void main() {
int n=6,y=1;
while(n) y*=n--;
printf("y=%d\n",y);
}
```

答案: ①y=720

```
61、#include<stdio.h> void main () {
#include<stdio.h>
void main() {
int i,s=0;
for(i=1;i<=5;i++) s+=i*i;
printf("s=%d\n",s);
}
```

答案: s=55

```
62、#include<stdio.h> void main () {
#include<stdio.h>
void main() {
int a[8]={12,39,26,41,55,63,72,40};
int i, i1=0, i2=0;
for(i=0;i<8;i++)
if(a[i]%2==1) i1++; else i2++;
printf("%d %d\n",i1,i2);
}
```

答案: ①4 4

```
63、#include<stdio.h> void main () {
#include<stdio.h>
void main() {
char a[]="abcdbfbgacd";
int i1=0, i2=0, i=0;
while(a[i]) {
if(a[i]=='a') i1++;
if(a[i]=='b') i2++;
i++;
}
printf("%d %d %d\n",i1,i2,i);
}
```

答案: ①2 3 11

```
64、#include<stdio.h> void main () {
#include<stdio.h>
void main() {
int a[3][4]={{1,2,7,8},{5,6,10,6},{9,12,3,4}};
int m=a[0][0];
int i,j;
```

```
for(i=0; i<3; i++)
for(j=0; j<4; j++)
if(a[i][j]>m) m=a[i][j];
printf("%d\n",m);
}
```

答案: ①12

```
65、#include<stdio.h> void main () {
#include<stdio.h>
void main() {
int a[8]={3,5,7,9,2,3,4,8};
int s=0,*p;
for(p=a;p<a+8;) s+=*p++;
printf("s=%d\n",s);
}
```

答案: s=41

```
66、#include<stdio.h> void main () { char a []="acdbfkmd"
```

```
26. #include<stdio. h>
void main() {
char a[]="acdbfkmd";
int i1=0, i2=0, i=0;
while(a[i]) {
if(a[i]<'d') i1++; else i2++;
i++;
}
printf("%d%d\n",i1,i2);
}
}
```

输出结果:

答案: 3 5

```
67、#include<stdio.h> void main () { char a[]
="acdbfdkmdaceg"
```

```
26. #include<stdio. h>
void main(){
char a[]="acdbfdkmdaceg";
int i=0,k=0;
while(a[i])
if(a[i++]=='d')k++;
printf("%d\n",k);
}
```

输出结果:

答案: 3

```
68、#include<stdio.h> void main () { char a[]="abcdbfkgamd";
```

```
3. #include<stdio. h>
void main() {
char a[]="abcdbfkgamd";
int i1=0, i2=0, i=0;
while(a[i]) {
if(a[i]<'e') i1++; else i2++;
i++;
}
printf("%d %d\n",i1,i2);
}
}
```

答案: 7 4

```
69、#include<stdio.h> void main () { int a,b;
```

```
#include<stdio. h>
void main()
{
    int a,b;
    for(a=2,b=3; a<15;) {
        printf("%d %d ",a,b);
        a=a+b;
        b=a+b;
    }
}
```

答案: 2 3 5 8 13 21

70、#include<stdio.h> void main () { int a[10]={12,39,26,41,55,63,72,40,83,95}; #include<stdio.h> void main() { int a[10]={12,39,26,41,55,63,72,40,83,95}; int i, i1=0, i2=0; for(i=0;i<10;i++) if(a[i]%2==1) i1++; else i2++; printf("%d %d\n",i1,i2); }

运行结果: 6 4

71、#include<stdio.h> void main () { int a[3][3]={{3,5,7},{9,11,13},{6,8,20}}; #include<stdio.h> void main() { int a[3][3]={{3,5,7},{9,11,13},{6,8,20}}; int i,\*p=&a[0][0]; for(i=0;i<9;i++) { if(\*p>10) printf("%d ",\*p); p++; } printf("\n"); }

运行结果: 11 13 20

72、#include<stdio.h> void main () { int a[6]= (36,25,48,24,55,40);

```
25. #include<stdio. h>
void main(){
    int a[6]={36,25,48,24,55,40};
    int i=0,x=24;
    while(a[i]!=x && i<6)i++;
    if(i<6)printf("%d\n",i);
}
```

输出结果: 4

答案: 3

73、#include<stdio.h> void main () { int a[8]= (12,39,26,41,55,63,72,40);

```
#include<stdio. h>
void main() {
    int a[8]={12,39,26,41,55,63,72,40};
    int i, c1=0, c2=0;
    for(i=0; i<8; i++)
        if(a[i]%2==1)c1++; else c2++;
    printf("%d,%d\n",c1,c2);
}
```

输出结果:

答案: 4,4

74、#include<stdio.h> void main () { int a[8]= [4,8,12,16,20,24,28,32];

```
27. #include<stdio. h>
void main(){
    int a[8]={4,8,12,16,20,24,28,32};
    int *p=a,s=0;
    do {s+=*p;p+=3;}while(p<a+8);
    printf("s=%d\n",s);
}
```

输出结果:

答案: s=48

75、#include<stdio.h> void main () { int a[8]=(4,8,12,16,20, 24, 28,32);

```
4. #include<stdio. h>
void main() {
    int a[8]={4,8,12,16,20,24,28,32};
    int *p=a;
    do {
        printf("%d ", *p);
        p+=3;
    } while(p<a+8);
    printf("\n");
}
```

答案: 4 16 28

76、#include<stdio.h> void main () { int a[8]=(76,63,54,95,40,75,90,66);

```
3. #include<stdio. h>
void main() {
    int a[8]={76,63,54,95,40,75,90,66};
    int i, s=0;
    for(i=0; i<8; i++)
        if(a[i]>=70 && a[i]<=90) s+=a[i];
    printf("s=%d\n",s);
}
```

答案: s=241

77、#include<stdio.h> void main () { int a[8]=17.6,4.5,8.3,9,12);

```
26. #include<stdio. h>
void main() {
    int a[8]={7,6,4,5,8,3,9,12};
    int i, s=0;
    for(i=0; i<8; i++)
        if(a[i]>=6 && a[i]<=9) s+=a[i];
    printf("s=%d\n",s);
}
```

答案: s=30

```
78. #include<stdio.h> void main () { int
a[8]={70,63,54,95,40,75,90,66};
#include<stdio.h>
```

```
void main() {
    int a[8]={70,63,54,95,40,75,90,66};
    int i, s=0;
    for(i=0; i<8; i++)
        if(a[i]>=70 && a[i]<=90) s+=a[i];
    printf("s=%d\n",s); }
```

运行结果: s=235

```
79. #include<stdio.h> void main () { int a[9]=
(36,25,48,24,55,40,18,66,20);
```

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

```
void main(){
    int a[9]={36,25,48,24,55,40,18,66,20};
    int i=0, k=-1, x=40;
    while(a[i]! =x && i<9)i++;
    if(a[i]==x) k=i;
    printf("k=%d\n",k);
}
```

输出结果:

答案: k=5

```
80. #include<stdio.h> void main () { int
a[9]={36,25,48,24,55,40,18,66,20};
```

```
#include<stdio.h>
void main() {
    int a[9]={36,25,48,24,55,40,18,66,20};
    int i, b1, b2;
```

```
b1=b2=a[0];
for(i=1; i<9; i++) {
    if(a[i]>b1) b1=a[i];
    if(a[i]<b2) b2=a[i]; }
printf("%d %d\n",b1,b2); }
```

运行结果: 66 18

```
81. #include<stdio.h> void main () { int a, b;
```

```
#include<stdio.h>
void main() {
    int a, b;
    for(a=2, b=3; b<20;) {
        printf("%d %d ", a, b);
        a=a+b;
        b=a+b; }
    printf("%d %d\n", a, b); }
```

运行结果: 2 3 5 8 13 21

```
82. #include<stdio.h> void main () { int f1,f2,i;
```

```
#include<stdio.h>
void main() {
    int f1,f2,i;
    f1=1;
    printf("%d ",f1);
    for(i=2;i<=5;i++) {
        f2=3*f1+1;
        printf("%d ",f2);
        f1=f2; }
    printf("\n"); }
```

运行结果: 1 4 13 40 121

```
83. #include<stdio.h> void main () { int i, s1=0, s2=0;
```

```
#include<stdio.h>
void main() {
    int i, s1=0, s2=0;
    for(i=0; i<10; i++)
        if(i%2) s1+=i;
        else s2+=i;
    printf("%d %d\n",s1,s2); }
```

运行结果: 25 20

```
84. #include<stdio.h> void main () { int i, s=0;
```

```
答案: s=20
85. #include<stdio.h> void main () { int i, s=0;
#include<stdio.h>
void main() {
    int i, s=0;
    for(i=1; i++) {
        if(s>30) eak;
        if(i%2==0) s+=i; }
    printf("s=%d\n",s); }
```

运行结果: s=42

```
86. #include<stdio.h> void main () { int i,j,k=0
```

答案: 10

```
87. #include<stdio.h> void main () { int i,j,k=0;
```

```
#include<stdio.h>
void main() {
    int i,j,k=0;
    for(i=0; i<5; i++)
        for(j=i; j<5; j++) k++;;
    printf("%d\n",k); }
```

运行结果: 15

```
88. #include<stdio.h> void main () { int i,s=0;
```

```
#include<stdio. h>
void main() {
    int i,s=0;
    for(i=1; i<6; i++) s+=i * i;
    printf("s=%d\n",s);
}
```

答案: s=55

```
89. #include<stdio.h> void main () { int i,s=0;
```

```
25. #include<stdio. h>
void main()
{
    int i,s=0;
    for(i=1; i<15; i++)
        if(i%4==0) s+=i;
    printf("s=%d\n",s);
}
```

输出结果:

答案: =24

```
90. #include<stdio.h> void main () { int i,s=0;
```

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

void main(){
    int i,s=0;
    for(i=1;s<10;i++)
        if(i%2==0)s+=i;
    printf("s=%d\n",s);
}
```

输出结果:

答案: s=12

```
91. #include<stdio.h> void main () { int i;
```

```
#include<stdio. h>
#include<string. h>
void main() {
    int i;
    unsigned int len=0;
    char * a[5]={"student","worker","cadre","soldier","n123"};
    for(i=0; i<5; i++)
        if(strlen(a[i])>len) len=strlen(a[i]);
    printf("len=%d\n",len);
}
```

答案: len=7

```
92. #include<stdio.h> void main () { int i, s=0;
```

```
#include<stdio.h>
void main() {
    int i, s=0;
    for(i=1;i<6;i++) s+=i*i;
    printf("s=%d\n", s); }
```

运行结果: s=55

```
93. #include<stdio.h> void main () { int n=4, y=0
```

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

void main() {
    int n=4, y=0;
    while(n-->0) y+=3;
    printf("y=%d\n",y);
}
```

输出结果:

答案: y=12

```
94. #include<stdio.h> void main () { int n=6, y=1;
```

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

void main() {
    int n=6, y=1;
    while(n-->0) y+=3;
    printf("y=%d\n",y);
}
```

答案: y=19

```
95. #include<stdio.h> void main () { int x=20
```

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

void main() {
    int x=20;
    int i=2;
    while(i<= x) {
        if(x%i==0) {printf("%d ", i); x/=i;}
        else i++;
    }
}
```

答案: 2 2 5

```
96. #include<stdio.h> void main () { int x=20
```

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

void main() {
    int x=20;
    int i=2;
    while(i<x) {
        if(x%i==0) {printf("%d ",i); x/=i;}
        i++;
    }
}
```

答案: 25

```
97. #include<stdio.h> void main () { int x=5;
```

```
#include<stdio.h>
void main() {
    int x=5;
    switch(2*x-1) {
        case 4: printf("%d",x); eak;
        case 7: printf("%d",2*x); eak;
        case 10: printf("%d",3*x); eak;
        default: printf("%s","default"); }
    printf("%s\n","switch end."); }
```

运行结果: default switch end

```
98. #include<stdio.h> void main () { struct Worker
a[4]={{\ "abe",20,2400}, {\ "defa",50,3600},
```

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

struct Worker { char name[15]; int age; double pay;};

void main() {
    struct Worker a[4]={{\ "abc",20,2400}, {\ "defa",50,3600},
        {\ "ghin",40,1800}, {\ "jkl",36,2200}};

    int i; double s=0;
    for(i=0; i<4; i++) s+=a[i]. pay;
    printf("s=%d. 2Lf\n",s);
}
```

输出结果:

答案: s=1000.00

```
99. #include<stdio.h> void main () {int i, s=0;
```

25. #include<stdio. h>

void main()

```
{
    int i, s=0;
    for(i=1;i<8;i++)
        if(i%2==0 || i%3==0) s+=i;
    printf("s=%d\n",s);
}
```

输出结果:

答案: s=15

100. #include<stdio.h> void main () {int i,j,k=0

1. #include<stdio. h>

void main() {

int i,j,k=0;

for({i=0; i<5; i++)

for(j=i; j<5; j++) k++;

printf("%d\n",k);

}

答案: 15

101. #include<stdio.h> void main () {int i,s=0;

24. #include<stdio. h>

void main() {

int i, s=0;

for(i=1;i<10;i+=2) s+=i;

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

}

输出结果:

答案: s=25

102. #include<stdio.h> void main () { int a[3][3]={13, 18, 16}

#include<stdio.h>

void main () {

int a[3][3]={{13, 18, 16}, {15, 19, 12}, {8, 15, 22}};

int i, j, x=a[0][0];

for (i=0; i<3; i++) for (j=0; j<3; j++) if (a[i][j]<x) x=a[i][j];

printf ("x=%d\n", x) ;

运行结果是: x=8

103. #include<stdio.h> void SB(char ch) {

答案: 123

104. #include<stdio.h> void SB(char ch) { switch(ch) {

#include<stdio.h>

void SB(char ch) {

switch(ch) {

case 'A': case 'a':

printf("WW"); eak;

case 'B': case 'b':

printf("GG"); eak;

default:

printf("BB"); eak; } }

void main() {

char a1='a',a2='B',a3='f';

SB(a1);SB(a2);SB(a3);

printf("\n"); }

运行结果: WW GG BB

105. #include<stdio.h> void SB(charch)

106. #include<stdio.h> Voidmain ()。 Intx=5,y=15, z=12;

执行下面程序的输出结果为 ( )。

#include<stdio.h>

Voidmain ()。

Intx=5,y=15, z=12;

if (x>y&& x>z) printf (" %d\n ",x)。

elseif (y>x&& y>z) printf (" %d\n ",y)。

elseif ( " %d\n ",z)。

)

答: 15

107. #include<stdio.h> void main ()。 int a[3][3]

执行下面程序的输出结果为 ( )。

#include<stdio.h>

void main ()。

int a[3][3]={{(3,8,6) , (5,9,2) , (8,5,12)}};

int i,j,s=0;

for (i=0; i<3; i++)。

for (j=0; j<3; j++)。

if (a[i][j]>6) s+=a[i][j];

printf (" s=%d\n ", s)。

)

答: s=37

108. #include<stdio.h> void main ()。 int i=0,s=0;

执行下面程序的输出结果为 ( )。

#include<stdio.h>

void main ()。

int i=0,s=0;

while (s<30) (i+=2; s+=i\*i;)

printf (" i=%d\n ",i)。

)

答: i=6

109. #include void main(){ int a[3][3]={{3,8,6}, {5,9,2},{8,5,12}};

执行下面程序的输出结果为( )。

答案: s=30

110. #includevoid main(){int a[8]={10,8,16,15,9,21,7,16};

答案: x=7

111. #include # define N 8 void main(){

# include<stdio. h>

# define N 8

void main(){

int a[N]= {3, 5, 8, 10, 15, 20, 32, 66};

int l, x;

for(i=0; i<N/2; i++) {

x=a[i]; a[i]=a[N-1-i]; a[N-1-i]=x;

}

printf(" %d\n" ,a[1]+a[3]);

}

执行下面程序的输出结果为( )。

答案: 47

112. #include void main () { int a[3][3]={{2 , 3 , 4}, {5 , 6 , 8}, {1 , 5 , 2}};

# include<stdio. h>

void main () {

int a[3][3]= {{2 , 3 , 4}, {5 , 6 , 8}, {1 , 5 , 2}};

int \* p= (int \*)a;

int s=0;

```
while( p< (int *)a+9) s+= *p ++;
printf(" %d\n" ,s);
}
```

执行下面程序的输出结果为()。

答案: 36

113、**#include void main () { int x=5 ,y=0;**

**# include<stdio. h>**

```
void main () {
    int x=5 ,y=0;
    switch(2 * x-3) {
        case 4: y +=x; eak;
        case 7: y+=2 * x+1; eak;
        case 10: y+=3 * x; eak;
    }
    printf("y= %d\n" ,y);
}
```

执行下面程序的输出结果为 ( )。

答案: y=11

114、**#include void main () {int a[3][3]={ {3 , 8 ,6} ,{ 5 ,9 ,2} , {8 , 5 ,12} } ;**

**# include<stdio. h>**

```
void main () {
    int a[3][3]={ {3 , 8 ,6} ,{ 5 ,9 ,2} , {8 , 5 ,12} } ;
    int i ,j ,s=0;
    for(i=0;i<3;i++)
        for(j=0;j<3;j++)
            if( a[i][j]>6)s+=a[i][j];
    printf("s= %d\n" ,s);
}
```

执行下面程序的输出结果为()。

答案: s=37

115、**#include void main () {int a[3][3]={ {3 , 8 ,6} , {5 , 9 , 2} , {8 , 5 ,12} } ;**

**# include<stdio. h>**

```
void main () {
    int a[3][3]={ {3 , 8 ,6} , {5 , 9 , 2} , {8 , 5 ,12} } ;
    int i ,j ,s=0;
    for(i=0;i<3;i ++)
        for(j =0 ;j<3 ;j++)
            if(a[i][j]%3= =0) s+=a[i][j] ;
    printf( "s= %d\n" ,s);
}
```

执行下面程序的输出结果为()。

答案: s=30

116、**#include void main () {int a[8]={10 , 8 , 16 , 15 , 9 , 21 , 7 ,16};**

**# include<stdio. h>**

```
void main () {
    int a[8]={10 , 8 , 16 , 15 , 9 , 21 , 7 ,16};
    int i ,x=a[0];
    for(i=1;i<8;i ++)
```

```
if(a[i]<x) x=a[i];
printf("x= %d\n" ,x);
}
```

执行下面程序的输出结果为()。

答案: x=7

117、**#include Void main() { Int a[8]={2,5,6,8, 4, 9, 3,6};**

**# include<stdio. h>**

```
Void main() {
    Int a[8]={2,5,6,8, 4, 9, 3,6};
    Int i, x=a[0]+a[1];
    for (i=1; i<7; i++) {
        int y=a[i]+a[i+1];
        if(y>x)x=y;
        printf("%d\n",x);
    }
}
```

答案: 14

118、**#include void main() {int i=1 ,s=0,**

**# include<stdio. h>**

```
void main() {
    int i=1 ,s=0,
    do {s+=i*i; i++;} while(s<30);
    printf( "s= %d\n" ,s);
}
```

执行下面程序的输出结果为()。

答案: s=30

119、**#include Void main(){ inti, f=1;**

**# include<stdio. h>**

```
Void main(){
    inti, f=1;
    for (i=1; i<4; i++) f=3*f+1;
    printf("f=%d\n", f);
}
```

答案: f=40

120、**#include void main () {int i=0 , s=0;**

**# include<stdio. h>**

```
void main () {
    int i=0 , s=0;
    while(s<30) {i+=2; s+=i * i; }
    printf("s= %d\n" ,s);
}
```

执行下面程序的输出结果为()。

答案: s=56

121、**#include void main() { int a[3][3]={ {2,3,4},(5,6,8), (1,5,2)} ;**

**# include<stdio. h>**

```
void main() {
    int x=5 ,y=0;
    switch(2*x-3) {
```

执行下面程序的输出结果为()。

答案: 36

122、**#include void main () { int x=5 ,y=0;**

**#include<stdio. h>**

```
void main() {
    int x=5 ,y=0;
    switch(2*x-3) {
```

```
case 4: y+=x; eak;
case 7: y+=2*x+1; eak;
case 10: y+=3*x; eak;
}
```

```
printf("y= %d\n" ,y);
}
```

答案: y=11

123、**#include void main () { int x=5,y=0;**

**# include<stdio. h>**

```
void main() {
    int x=5 ,y=0;
    switch(2 * x+3) {
        case 4: y += x; break;
        case 7: y += 2 * x + 1; break;
        case 10: y += 3 * x; break;
        default: y += x - 1;
    }
    printf("y= %d\n" ,y);
}
```

答案: y=4

124、**#include #define M6**

**#include<stdio.h>**

**#define M6**

**vold main()**

```
{
    inti, x=0
    inta [M]
    for(i=0;
    = {2, 5, 8, 4, 6, 9};
    i<M;i ++ )x +=a[i];
    printf(" %d\n", x);
}
```

输出结果:

答案: 34

125、**#include #define NS**

**#include<stdio.h>**

**#define NS**

```
void main(){
    inta[N]={3, 5, 8, 10, 15, 20, 32, 66};
    Intl, x;
    for(i=0;i<N/2;i++){
        x=a[i];a [i]=a[N - 1 - i] ;a [N - 1 - i]=x;
    }
    printf("%d\n", a [1]+a[3]);
```

答案: 47

```
126、#include #define N5 void main () {
#include<stdio.h>
#define N5
void main () {
int a[N]
int i;
a[0]=1
for (i=1; i<N; i++) a[i]=2*a[i-1]+1;
printf ("%d\n", a[N-1]);
}
```

执行下面程序的输出结果为 ( )

答案: 31

```
127、#include #void main () { inti, fl=1, f2;
#include<stdio.h>
#void main(){
inti, fl=1, f2;
for(i=2;i<6;i++)
f2=3*fl
fl=f2
printf("fZ=%d\n", f2);
}
```

答案: f2=81

```
128、#include struct Worker{char name[15];int age;float pay;
#include<stdio.h>
struct Worker{char name[15];int age;float pay;
void main(){
struct Worker x={"wanghua", 48, 2500};
struct Workery, *P;
y=x;p=&x
printf("%d\n", y.age+p ->age);
}
```

输出结果:

答案: 96

```
129、#include void main () { inta~25, b=18;
```

执行下面程序的输出结果为(

```
#include<stdio.h>
void main(){
inta~25, b=18;
if(a>b)printf("%S\n", "a>b");
if(a==b)printf("%s\n", "a==b");
if(a<b)printf("%s\n", "a<b");
答案: a>b
```

```
130、#include void main () { inta[8]={3, 8, 6, 5, 9, 2, 8, 5};
#include<Stdio.h>
void main(){
inta[8]={3, 8, 6, 5, 9, 2, 8, 5};
inti, s=0;
for(i=0;i<8;i++)
if(i%2)s+=a [i];
printf("S=%d\n", s);
答案: s=20
```

```
131、#include void main () { inta (8)={10, 8, 16, 15, 9, 21,
7, 16};
#include<stdio.h>
void main(){
inta (8)={10, 8, 16, 15, 9, 21, 7, 16};
inti, x=a (0) ;
for(i=1;i<8;i++)
if(a (i) <x)x=a (i) ;
printf("x=%d\n", x);
}
```

答案: x=7

```
132、#include void main () { intn=0, y=0;
#include<stdio.h>
void main(){
intn=0, y=0;
while(n<10){n++, y+=n
printf("y=%d\n", y);
}
```

答案: y=55

```
133、#include void main () {inta (8)={12, 15, 20, 8, 5, 9,
16, 10};
#include<stdio.h>
void main(<
inta (8)={12, 15, 20, 8, 5, 9, 16, 10};
inti, j, x=a[o] +a (1) ;
for(i=0;i<8;i++)
for(j=i+1;j<8;j++)
if(a[i] +a (j) >x)x=a (i) +a (j) ;
printf("%d", x);
}
```

答案: 36

```
134、#include # define N 8
```

```
# include<stdio. h>
# define N 8
void main() {
int a[N]={3,5,8,10,15,20,32,66};
int i, x;
for(i=0; i<N/2; i++) {
x=a[i]; a[i]=a[N-1-i]; a[N-1-i]=x;
}
printf("%d\n",a[1]+a[3]);
}
```

答案: 47

```
135、#include #define N 5 void main () {
#include<stdio. h>
#define N 5
void main() {
int a[N];
int i;
a[0]=1; a[1]=2;
for(i=2; i<N; i++) a[i]=3 * a[i-2]+a[i-1]+1;
printf("%d\n",a[N-1]);
}
```

答: 32

```
136、#include void main () { int x=12,s=0;
#include<stdio. h>
void main() {
int x=12,s=0;
do {s+=x; x-=3;} while(x>0);
printf("s= %d\n",s);
}
```

答案: s=30

```
137、#include void main(){inti=0, s=0;
```

```
# include<stdio. h>
void main() {
int i=0, s=0;
while(s<30) {i+=2; s+=i * i;}
printf("s= %d\n",s);
}
```

答案: s=56

```
138、#include void main () { int a[8]= {3,8,6,5,9,2,8,5};
```

```
#include<stdio. h>

void main() {

    int a[8]={3,8,6,5,9,2,8,5};

    int i,s=0;

    for(i=0;i<8;i++)

        if(a[i]>5) s+=a[i];

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

}
```

答: s=31

139、#include void main() { int a[8] = { 10, 8, 16, 15, 9, 21, 7, 16};

```
#include<stdio. h>

void main() {

    int a[8]={10,8,16,15,9,21,7,16};

    int i,x=a[0];

    for(i=1;i<8;i++)

        if(a[i]<x) x=a[i];

    printf("x=%d\n",x);

}
```

答: x=7

140、#include void main () { int i,f1=1,f2;

```
#include<stdio. h>

void main( ) {

    int i,f1=1,f2;

    for(i=2;i<5;i++) {

        f2=3 * f1;

        f1=f2;

    }

    printf("f2=%d\n",f2);

}
```

执行下面程序的输出结果为 (f2=27)

141、#include void main () { int x=5 ,y=8;

```
#include<stdio. h>

void main() {

    int x=5,y=8;

    if(x>=y) printf("%d %d\n",x,y);

    else printf("%d %d\n",y,x);

}
```

答: 8 5

142、#include void main () { int x=5,y=0;

```
#include<stdio. h>

void main() {

    int x=5,y=0;

    switch(2 * x+3) {

        case 4: y+=x; break;

        case 7: y+=2 * x+1; break;

        case 10: y+=3 * x; break;

        default: y+=x-1;

    }

    printf("y=%d\n",y);

}
```

答: y=4

143、#include void main () { int x=5,y=8,z=4;

```
#include<stdio. h>

void main( ) {

    int x=5,y=8,z=4;

    if(x>y) x=y;

    if(x>z) x=z;

}
```

执行下面程序的输出结果为 (4)

144、#include void main () { inti, s1=0, s2=0;

```
#include<stdio. h>

void main() {

    int i, s1=0, s2=0;

    for(i=1;i<10;i++)

        if(i%2) s1+=i;

        else s2+=i;

    printf("%d %d\n",s1,s2);

}
```

答: 25 20

145、#include void main () { intx=1,y;

```
#include<stdio. h>

void main() {

    int x=1,y;

    do {y=2 * x+1; x=y;} while(y<30);

    printf("y=%d\n",y);

}
```

答: y=31

146、#include void main () { int a=5, b=8, c=6, x;

```
#include<stdio.h>

void main () {

    int a=5, b=8, c=6, x;

    if (a>b) x=a; elsex=b;

    if (c>x) x=c

    printf ("%d\n", x);

}
```

执行下面程序的输出结果为 ( )

答案: 8

147、#include void main () { int a[4][3]={ (3, 8, 6), {5, 9, 2}, {8, 5, 12}, {5, 13, 10}};

```
#include<stdio.h>

void main () {

    int a[4][3]={ (3, 8, 6), {5, 9, 2}, {8, 5, 12}, {5, 13, 10}};

    inti,j,s=0;

    for (i=0; i<4; i++)

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

    if (a[i][j]>8) s+=a[i][j];

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

}
```

执行下面程序的输出结果为 ( )

答案: s=44

148、**#include void main () { int x=5, y=8, z=6, w=0;**

```
#include<stdio.h>
void main(){
    int x=5, y=8, z=6, w=0;
    if(x>=y) w=x; else w=y;
    if(w<z)w=z;
    printf("w=%d\n", w);
}
```

答案: w=8

149、**#include void main () { inta[6] ={36, 25, 48, 24, 55, 40};**  
**#include<stdio.h>**

```
void main(){
    inta[6] ={36, 25, 48, 24, 55, 40};
    int i=0, x=24;
    while(a [i] !=x&& i<6)i++;
    if(i<6)printf("%d\n", i);
}
```

输出结果:

答案: 3

150、**#include void main () { inti, s=0;**

```
#include<stdio.h>
void main(){
    inti, s=0;
    for(i=1;i<10;i++)
        if(i%2==0)s+=
        printf("s=%d\n", s);
}
```

输出结果:

答案: s=20

151、**#include void SB(char ch){**

```
#include<stdio.h>
void SB(char ch){
    switch(ch){
        case'A':printf("1");eak;
        case'B':printf("2");eak;
        default:printf("3");eak;
    }
}
void main(){
    char al='a', aZ 一'B', a3='A';
    SB(al);SB(aZ);SB(a3);
}
```

输出结果:

答案: 321

152、**#include voidmain () { int n=2, y=0; while (n<10) {y+=n;**  
**n+=2; }**

```
#include<stdio.h>
voidmain () {
    int n=2, y=0; while (n<10) {y+=n; n+=2; }
    printf ("y=%d\n", y);
}
```

执行下面程序的输出结果为 ( )。

答案: y=20

153、**#include void main () 。**  
执行下面程序的输出结果为 ( )。

```
#include<stdio.h>
void main ()。
int i=0,s=0;
while (s<30) (i+=2; s+=i*i);
printf ( " i=%d\n " ,i)。
)
```

答: i=6

154、**include<stdio. h>void main(**

执行下而程序的输出结果为 (x=27)

155、**include<stdio.h> void main () { int x=5**

24. **#include<stdio. h>**

```
void main()
{
    int x=5;
    switch(2 * x-1){
        case 4: printf("%d", x); break;
        case 7: printf("%d", 2 * x); break;
        case 10: printf("%d", 3 * x); break;
        default: printf("%s", "default");
    }
}
```

输出结果:

答案: default

156、**include<stdio.h> void main () {int fl=1,f2,i**

25. **#include<stdio. h>**

```
void main() {
    int f1=1,f2,i;
    printf("%d", f1);
    for(i=1;i<4;i++) {
        f2=3 * f1+1;
        printf("%d", f2);
        f1=f2;
    }
}
```

输出结果:

答案: 1 4 13 40

157、**include<stdio.h>defineN6**

2. **#include<stdio. h>**

**#define N 6**

**void main()**

```
{
    int i, a[N]={2,5,8,10,15,21};
    for(i=0; i<N; i++)
        if(a[i]%5) printf("%d", a[i]);
    printf("\n");
}
```

答案: 2 8 21

158、**include<stdio.h>intWF(intx,inty){**

```
#include<stdio. h>
int WF(int x, int y){
    x=x+y;
    Y+=x;
    return x+ y;
}
void main(){
    int x= 3, y=5;
    printf ("%d\n", WF(x, y));
}
```

答案: 21

159、`include<stdio.h>void main () { int a,b;`

```

4. #include<stdio. h>

void main()
{
    int a,b;
    for(a=2,b=3; b<20;) {
        printf("%d %d",a,b);
        a=a+b;
        b=a+b;
    }
    printf("%d %d\n",a,b);
}

```

答案: 2 3 5 8 13 21

160、`include<stdio.h>void main () { int i;`

```

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

void main() {
    int i;
    unsigned int len;
    char * a[5]={"student","worker","cadre","soldier","zzeasan123"};
    len=strlen(a[0]);
    for(i=1; i<5; i++)
        if(strlen(a[i])>len) len=strlen(a[i]);
    printf("%d\n",len);
}

```

答案: 10

161、`include<stdio.h>void main () { int i,j,k=0;`

```

#include<stdio. h>
void main(){
    int i,j,k=0;
    for(i=0;i<5;i++)
        for(j=i; j<5;j++)k++
    printf ("%d\n", k);
}

```

答案: 15

162、`include<stdio.h>voidLE(inta,intb){`

26 15

163、`include<stdio.h>voidmain () {int i,s=0;`

答案: s=55

164、`include<stdio.h>[intWF(intx,inty){`

```

#include<stdio. h>
int WF(int x, int y){
    x=x+y;
    Y+=x;
    return x+ y;
}
void main(){
    int x= 3, y=5;
    printf ("%d\n", WF(x, y));
}

```

答案: 21

165、`includeconstintM=20`

```

#include
const int M=20;
void main()
{
    int i,c2,c3,c5;
    c2=c3=c5=0;
    for(i=1; i<=M; i++) {
        if(i%2==0) c2++;
        if(i%3==0) c3++;
        if(i%5==0) c5++;
    }
    printf("%d %d %d\n",c2,c3,c5);
}

```

答案: 10 6 4

166、`includeinta[10]={4,5,6,15,20,13,12,7,8,9}`

```

#include
int a[10]={4,5,6,15,20,13,12,7,8,9};
void main()
{
    int i,s[4]=0;
    for(i=0; i<10; i++) {
        switch(a%4) {
            case 0: s[0]+=a;eak;
            case 1: s[1]+=a;eak;
            case 2: s[2]+=a;eak;
            default: s[3]+=a;eak;
        }
    }
    for(i=0; i<4; i++) printf("%d ",s);
}

```

答案: 4 4

167、`includevoidfun(charss[])`

```

#include
void fun(char ss[]);
void main() {
    char s[15]="0123456789";
}

```

```

fun(s);
printf("%s\n",s);
}

```

```

void fun(char ss[]) {
    int i, n=strlen(ss);
    for(i=0; i<n/2; i++) {
        char c=ss;
        ss=ss[n-1-i];
        ss[n-1-i]=c;
    }
}

```

答案: 9876543210

168、`includevoidmain ()`

```

#include
void main()
{
    int x=5;
    switch(2*x-3) {
        case 4: printf("%d ",x);
        case 7: printf("%d ",2*x+1);
        case 10: printf("%d ",3*x-1); eak;
        default: printf("%s","default\n");
    }
}

```

答案: 11 14

169、`includevoidmain ()`

```

#include
void main()
{
    int i,s;
    for(i=1,s=0;i<10;i++) {
        if(i%3==0) continue;
        s+=i;
    }
    printf("%d\n",s);
}

```

答案: 27

170、`includevoidmain () {`

```

#include
void main() {
    int i, s1=0, s2=0;
    for(i=0;i<10;i++)
        if(i%2) s1+=i; else s2+=i;
    printf("%d %d\n",s1,s2);
}

```

答案: 25 20

171、`includevoidmain () {`

```

#include
void main() {
}

```

```
int n=6,y=1;
while(n) y*=n--;
printf("y=%d\n",y);
}
```

答案: y=720

172、`includevoidmain () {`

```
#include
void main() {
int a[8]={12,39,26,41,55,63,72,40};
int i, i1=0, i2=0;
for(i=0;i<8;i++)
if(a%2==1) i1++; else i2++;
printf("%d %d\n",i1,i2);
}
```

答案: 4 4

173、`includevoidmain () {`

```
#include
void main() {
char a[]="abcdbfbgacd";
int i1=0, i2=0, i=0;
while(a) {
if(a=='a') i1++;
if(a=='b') i2++;
i++;
}
printf("%d %d %d\n",i1,i2,i);
}
```

答案: 2 3 11

174、`includevoidmain () {`

```
#include
void main() {
int a[3][4]={{1,2,7,8},{5,6,10,6},{9,12,3,4}};
int m=a[0][0];
int i,j;
for(i=0; i<3; i++)
for(j=0; j<4; j++) if(a[j]>m) m=a[j];
printf("%d\n",m);
}
```

答案: 12

175、`void main () { int a[3][3]=1(3,5,7),(9,11,13), (6,8,20);`

27. `#include<stdio. h>`

```
void main() {
int a[3][3]={{3,5,7},{9,11,13},{6,8,20}};
int i, k=0, * p=&a[0][0];
for(i=0;i<9;i++)
if(* p++<=8) k++;
printf("k=%d\n",k);
}
```

输出结果:↵

答案: k=5 ↵

176、按照“`int FF (inta[], intn)`”的函数声明, 编写出递归函数的定义, 求出数组 a 中所有 n 个元素之积并返回

```
35. int FF(int a[], int n)
{
if(n<=0) {printf("n 值非法\n"),exit(1);}
if(n==1) return a[0];
else return a[n-1] * FF(a,n-1);
}
```

177、当使用 `xkZC` 调用此函数时, 返回值为()

当使用 `xkZC` 调用此函数时, 返回值为() .

```
int xk2(int a, int b, int c){
if(a>=b&&a>=c)return a,
if(b>=a 且 b>=c)return b,
return c;
}
```

答案: 7

178、假定使用 `xxk3 (4)` 调用下面函数, 则返回值为 ()

```
int xxk3(int n) {
int i, s=1,
for(i=1;i<=n;i++)s*=i;
return s;
}
```

答案: 24

函数功能(26)--电大资源网: <http://www.dda123.cn/> (微信搜: 905080280)

- `#include<math.h>`
- `#include<stdio.h> int SA(int a, int b) {...`
- `double SF(double x, int n) { //n 为正整数...`
- `int * LI(int n)`
- `int Count(struct IntNode *f)`
- `int Count(struct IntNode *f)`
- `int FF(struct IntNode*f)`
- `int FH ()`
- `int FindMax(struct IntNode *f) //f 为一...`
- `int fun(int m, int n) {`
- `int i=2; if(x==2||x==3) return 1...`
- `int LK(double a[], int n) {`
- `int SC(int a,int b,int c)`
- `int SG(int x) { //x 为大于等于...`
- `int WB(int a[], int n, int x) {`
- `int WC(int a[], int n, int k) { int i, c=0...`
- `void Output(struct IntNode *f) //f 为单...`
- `void QA(struct Worker a[], int n) {`
- `void QA(struct Worker a[], int n) {int i;...`
- `void xw1(char* fname) {`
- `void xw1(char* fname) {`
- `voidQA(structWorkera[],intn){`
- 假定结点的结构类型为 `IntNode`, 它的数值域为 `d...`
- 假定结点的结构类型用 `IntNode` 表示, 它的数值...
- 假定结构类型 `struct Worker` 的定义体为: ...
- 假定使用 `SF (20)` 调用下面函数, 则得到的输出结果...

1、`#include<math.h>`

`#include<math.h>`

`int SG (int x) { //x 为大于等于 2 的整数`

`int a= (int) sqrt (x) ; //sqrt (x) 取 x 的平方根`

`int i=2;`

`while (i<=a) {`

`if (x%i==0) eak;`

`i++;`

`}`

`if (i<=a) return 0;else return 1;`

函数功能:

答案: 判断 x 是否为一个素数, 若是则返回

2、`#include<stdio.h> int SA(int a, int b) {`

`;`

函数功能:

答案: 比较两个整数 a 和 b 的大小, 若 `a>b` 则返回 1, 若 `a=b` 则返回 0, 若 `a<b` 则返回-1。

3、`double SF(double x, int n) { //n 为正整数`

`double SF(double x, int n) { //n 为正整数`

`double p=1,s=1;`

`int i;`

```

for(i=1;i<=n;i++) {
    p*=x;
    s+=p;
}
return s;
}

```

答案: 计算出  $1+x+x^2+\dots+x^n$  的值并返回。

#### 4、int \* LI(int n)

```

int *LI (int n) {
int i;
int *a=malloc (n*sizeof (int) );
for (i=0; i<n; i++) scanf ("%d", a+i) ;
return a;
}

```

函数功能:

答案: 首先建立一个动态整型数组 a[n], 接着从键盘上输入 n 个整数到 a[n] 中, 最后返回数组 a[n] 的首地址。

#### 5、int Count(struct IntNode \*f)

```

int Count(struct IntNode *f)
{ //f 为指向一个单链表的表头指针
int c=0;
while(f) {
c++;
f=f->next;
}
return c;
}

```

假定 struct IntNode 的类型定义为:

```
struct IntNode {int data; IntNode* next;};
```

答案: 统计出以表头指针为 f 的链表中结点的个数。

#### 6、int Count(struct IntNode \*f)

```
int Count(struct IntNode *f) { //f 为指向一个单链表的表头指针
```

```

int c=0;
while(f) {
c++;
f=f->next;
}
return c;
}

```

假定 struct IntNode 的类型定义为:

```
struct IntNode {int data; IntNode* next;};
```

答案: Count 函数可以统计单链表的结点个数, 其表头指针为 f

#### 7、int FF(struct IntNode \*f)

```

int FF (struct IntNode *f) //f 为一个单链表的表头指针
{
int c=0;
if(f==NULL)return 0;
while(f){c++;f=f->next;
}
return c;
}

```

函数功能:

答案: 求出并返回由 f 所指向的单链表中结点的个数。

#### 8、int FH ()

```

int FH() {
int x,y=0;
scanf("%d", &x);
while(x>0) (y+=x; scanf("%d", &x);)
return y;
}

```

答案: 求出从键盘上输入的一批整数的总和, 以小于等于 0 作为结束键盘输入的标志。

#### 9、int FindMax(struct IntNode \*f) //f 为一个单链表的表头指针

```

int FindMax(struct IntNode *f) //f 为一个单链表的表头指针
{
int x;
if(!f) {printf("单链表为空\n"),exit(1);}
x=f->data;
f=f->next;
while(f) {
if(f->data>x) x=f->data;
f=f->next; }
return x; }

```

假定 struct IntNode 的结点类型定义为:

```
struct IntNode { int data; struct IntNode* next;};
```

函数功能: 求出并返回由 f 所指向的单链表中所有结点的最大值。

#### 10、int fun(int m, int n) {

```

int fun(int m, int n) {
int c=0;
static int b=2;
if(m<b || n<b) return m*n;
else if(m%b==0 && n%b==0) {c=b; return
c*fun(m/b,n/b);}
else {b++; return fun(m,n);}
}

```

答案: 一个递归函数过程, 求出两个自然数 m 和 n 的最小公倍数。

#### 11、int i=2; if(x==2||x==3) return 1

```

int i=2;
if(x==2||x==3) return 1;
while(i*i<=x){
if(x%i==0) eak;
i++;
}
if (i * i<=x)return 0;else return 1;
}

```

函数功能:

答案: 判断 x 是否为一个素数, 若是则返回 1, 否则返回 0。

#### 12、int LK(double a[], int n) {

```

int LK(double a[], int n) {
double s=0;
int i,m=0;
for(i=0;i<n;i++) s+=a[i];
s/=n;
for(i=0;i<n;i++)

```

```

if(a[i]>=s) m++;
return m;
}

```

答案: 求出并返回数组 a 的 n 个元素中大于等于平均值的元素个数。

#### 13、int SC(int a,int b,int c)

```

int SC(int a, int b, int c) {
if(a>=b && a>=c) return a;
if(b>=a && b>=c) return b;
return c;
}

```

函数功能:

答案: 求出 a, b, c 三个数中的最大值并返回。

#### 14、int SG(int x) { //x 为大于等于 2 的整数

```

int SG(int x) { //x 为大于等于 2 的整数
int i=2;
while(i*i<=x) {
if(x%i==0) eak;
i++;
}
if(i*i<=x) return 0; else return 1;
}

```

答案: 判断 x 是否为一个质数 (素数), 若是则返回 1, 否则返回 0。

#### 15、int WB(int a[], int n, int x) {

```

int WB(int a[], int n, int x) {
for(int i=0;i<n;i++)
if(a[i]==x) return 1;
return 0;
}

```

答案: 从数组 a[n] 中顺序查找值为 x 的元素, 若查找成功则返回 1, 否则返回 0。

#### 16、int WC(int a[], int n, int k) { int i, c=0;

```

int WC(int a[], int n, int k) {
int i, c=0;
for(i=0; i<n; i++)
if(a[i]>=k) c++;
return c; }

```

函数功能: 统计并返回一维整型数组 a[n] 中大于等于 k 的值的个数。

#### 17、void Output(struct IntNode \*f) //f 为单链表的表头指针

```

void Output(struct IntNode *f) //f 为单链表的表头指针
{
if(!f) return;
while(f) {
printf("%d ",f->data);
f=f->next; }
printf("\n"); }

```

假定 struct IntNode 的类型定义为:

```
struct IntNode { int data; struct IntNode* next;};
```

函数功能: 遍历并输出由 f 所指向的单链表中每个结点的值。

#### 18、void QA(struct Worker a[], int n) {

```
void QA(struct Worker a[], int n) {
```

已整理 700 个国开科目, 有需要请直接微信 905080280, 说明要购买的试卷号及科目名称即可  
ps: 资料考前整理, 只供大家复习使用! 已和最新历届试题核对, 有新题并已整合, 以此版为准



手机用浏览器扫码访问电大资源网

```
int i;
for(i=1; i<n; i++)
    scanf("%s %d %f", &a[i].name, &a[i].age, &a[i].pay);
}
```

假定结构类型 struct Worker 的定义如下:

```
struct Worker { char name[15]; int age; float pay;};
```

答案: 从键盘输入 n 个 Worker 类型的记录到一维数组 a 中。

19、void QA(struct Worker a[], int n) {int i;

```
void QA(struct Worker a[], int n)
```

```
{int i;
for(i=0; i<n; i++)
    scanf("%s%d%f", a[i].name, &a[i].age, &a[i].pay);}
```

假定结构类型 struct Worker 的定义如下:

```
struct Worker{char name[15];int age;float pay;};
```

函数功能:从键盘上为具有 struct Worker 类型的数组 a[n]输入 n 个记录。

20、void xw1(char\* fname) {

```
void xw1(char* fname) {
```

```
FILE* fout=fopen(fname,"w");
```

```
char a[20];
```

```
printf("每个字符串长度小于 20, 字符串 end 作为结束标志\n");
```

```
while(1) {
```

```
scanf("%s",a);
```

```
if(strcmp(a,"end")==0) eak;
```

```
fputs(a,fout);
```

```
fputc('\n',fout);
```

```
}
```

```
fclose(fout);
```

```
}
```

答案: 把从键盘上输入的若干个字符串保存到由 fname 参数所指定的文本文件中。

21、void xw1(char\* fname) {

```
void xw1(char* fname) {
```

```
FILE* fout=fopen(fname,w);
```

```
char a[20];
```

```
printf(每个字符串长度小于 20, 字符串 end 作为结束标志\n);
```

```
while(1) {
```

```
scanf(%s,a);
```

```
if(strcmp(a,end)==0) eak;
```

```
fputs(a,fout);
```

```
fputc('\n',fout);
```

```
}
```

```
fclose(fout);
```

答案: xw1 函数可以输入若干字符串保存到 fname 所指定的文件中, 直至遇到字符串 end

22、void QA(struct Workera[],intn){

```
void QA(struct Worker a[], int n) {
```

```
int i;
```

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

```
scanf(%s %d %f,&A.name,&A.age,&A.pay);
}
```

假定结构类型 struct Worker 的定义如下:

```
struct Worker { char name[15]; int age; float pay;};
```

答: 从键盘输入 n 个 worker 类型的记录到一组数组 a 中。

23、假定结点的结构类型为 IntNode, 它的数值域为 data, 链接指针域为 next, 请指出下面函数的功能。

```
void QC (struct IntNode*f) {/f 为一个单链表的表头指针
```

```
while (f) {
```

```
printf ("%d", f->data) ;
```

```
f=f->next;
```

```
函数功能:
```

答案: 遍历输出由表头指针 f 所指向的单链表中每个结点的 data 域的值。

24、假定结点的结构类型用 IntNode 表示, 它的数值域为 data, 链接指针域为 next, 请根据下面函数定义写出相应的功能。

假定结点的结构类型用 IntNode 表示, 它的数值域为 data, 链接指针域为 next, 请根据下面函数定义写出相应的功能。

```
int FF(struct IntNode*f)/f 为指向一个单链表的表头指针
```

```
{
```

```
Int n=0
```

```
if(f==NULL)returno;
```

```
while(f){n++;f=f->next;}
```

```
returnlll;
```

```
}
```

```
函数功能:
```

答案: 求出并返回由 f 所指向的单链表中所含结点的个数。

25、假定结构类型 struct Worker 的定义体为:

```
{char name[15];int age; float pay;};
```

```
void QA (struct Worker a[], int n)
```

```
int i;
```

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

```
scanf ("%s%d%f", a[i].name, &a[i].age, &a[i].pay) ;函数功能:
```

答案: 从键盘上为具有 struct Worker 类型的数组 a[n]输入 n 个记录。

26、假定使用 SF (20) 调用下面函数, 则得到的输出结果为

假定使用 SF (20) 调用下面函数, 则得到的输出结果为 ( )

```
void SF (int x) {
```

```
int i=2; while (x>=i) {
```

```
if (x%i= =0) {printf ("%d", i) ;x/=i;}
```

```
i++;
```

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

```
答案: 2 5 2
```

上一次考试有 150 多个科目改版, 电大资源网每学期均会在期末考试前整合最新历届试题+形考作业+综合练习册题目, 有需要直接访问 <http://www.dda123.cn/>

任何问题都可以联系我微信: 905080280

请直接打印, 已按字母排版