一、**基础知识**

计算机工作原理

软硬件系统

进制转换

Windows 操作

网络

多媒体

病毒

二、类型及表达式

一）合法的标志符：

命名规则

（1）有效字符：只能由字母、数字和下划线组成，且以字母或下划线开头。

（2）有效长度：随系统而异，但至少前 8 个字符有效。

（3）C 语言的关键字不能用作变量名。

例题：

1．是合法的用户自定义标识符的(　　)

A)ah-spks　　B)double　　C)<ctrl>　　D)_myapple

2．以下正确的 C 语言标识符是（　　）

A）%x　　B)a+b　　C)a123　　D)test!

3．以下不能定义为用户标识符的是(　　)

A)　Void　　B)　b_2　　C)　int　　D)　name

二）合法的整型常量

（1）十进制。例如 10、36。

（2）八进制（以数字 0 开头）。例如 012。（注意 0 后的数字不能>=8）

（3）十六进制（以数字 0 +小写字母 x 开头）。例如 0x36。

例题：

1．以下选项中可作为 C 语言合法整数的是(　　)

A)　11010 B　　B)　0583　　C)　x2b2　　D)　0xafb

三）合法的实型常量

（1）十进制小数形式。例如 3.14、9.8。　（小数点后数字可省略）

（2）指数形式：

<尾数>E（e）<整型指数>。

字母 e 或 E 前后必须有数字,且后面指数必须为整数

例题：

1．以下选项中可作为 C 语言合法常量的是（　　）

A）-80.　　B）-080　　C）-8e1.0　　D）-80.0e

四）合法的字符型常量

（1）用一对单引号括起来的单个字符，称为字符常量。

（2）以“\ ”开头的转义字符

例题：

1．设有说明语句:char a='\123';则变量 a(　　)

A)包含 1 个字符　B)包含 2 个字符　C)3 个字符　D)说明不合法

2．C 语言中，"\x5d"在内存中占用的字节数是（　　）

A)　2　B)　5　C)　4　D)　1

3．字符串"\t\x42\\bcd\n"的长度是（　　）。

A)7　B)10　C)12　D)13

五）各种基本类型变量所占的字节数

int　2　long　4　unsigned　2

float　4　double　8　char　1

例题：

1．下列式中，值不为 4 的表达式是（　）

A）sizeof(unsigned long)　B)sizeof(long)　C)sizeof(unsigned int)　D)sizeof(float)

六）合法的表达式

（1）%两边操作数必须是整型

（2）赋值的左边一定是变量

例题：

1．设变量 a、b、c 已定义并赋值，则下列表达式中符合 C 语言语法规则的是（　）

A）a=5++　B)a=b=c++　C)a%=2　D)b=a+1=2

七）表达式的值

每个表达式都有值

（1）算术表达式的值就是计算结果

（2）赋值表达式的值被赋值变量的值

（3）逗号表达式的值是最后一个表达式的值

（4）注意两个表达式的意义：令 a 是数字字符，则 **a-'0'**就是对应的数字

令 a 是字母字符，则 a-32 就是小写变为大写，a+32 就是大写变为小写

例题：

1．　设整型变量 a 的值为 2，下列表达式值为 1 的是（　）

A）a%3　B)a/3　C)--a　D)a++

2．　下列表达式的值为 0 的是（　　）

A)　7/8　B) 7%8　C)　7/8.0　D) 7<8

3．　设 float m=4.0, n=4.0;　使 m 为 10.0 的表达式是（　　）

A)　m-=n*2.5　B)　m/=n+9　C)　m*=n-6　D)　m+=n+2

4．　逗号表达式（a=4*5,a*2）,a+15 的值是（　　）

A)　35　B)　40　C)　55　D)　20

八）++、--运算

1）先将表达式中++、--去掉

2）再将前置型的写在表达式上方

3）最后将后置型的写在表达式下方

例题：

当 i=4,j=5 时，表达式 3-(i++)*4+(--j)的值如何，i,j 的值多少？

九）表达式的类型

自动转换：表达式中有整型和字符型，结果一定为整型

表达式中有实型，结果一定为 double 类型

强制转换：可以强制转换为指定的类型

例题：

1. 下列式中，最终运算结果的数据类型不是双精度的表达式的是（　　）

A）(int)(3+3.0)　　B)1e-3　　C)(double)(3)　　D)(int)3.0+3.0

二、顺序结构

一）putchar 和 getchar：一次只能输入输出一个字符

例题：

1. putchar 函数可以向终端输出一个（　　）

A）整数　　B）实数　　C）字符串　　D）字符

二）printf 和 scanf 语句的使用方法：

scanf 注意数据输入时的分隔符（“”中有的要原样输入，没有的数值数据以空格、回车、Tab 键为默认分隔符，字符数据没有分隔符）

例题：

1. 设有语句 scanf(%d,%d”,&m,&n);要使 m、n 的值依次是 2、3，正确是输入是（　　）

A）2　3　　B）2，3　　C）2；3　　D）2

2. 设变量定义为 int a,b;执行下列语句时,输入(　　),则 a 和 b 的值都是 10

scanf(“%d,%d”,&a,&b);

A)10　10　　B) 10,10　　C)a=10　b=10　　D)a=10,b=10

3. 有以下程序

```
main()
{    int   m,n,p;
     scanf("m=%dn=%dp=%d",&m,&n,&p);
     printf("%d%d%d\n",m,n,p);
   }
```

若想从键盘上输入数据,使变量 m 中的值为 123,n 中的值为 456,p 中的值为 789,则正确的输入是（　　）

A）m=123n=456p=789　　B）m=123　n=456　p=789

C）m=123,n=456,p=789　　D）123　456　789

printf 注意输出形式%m.n 形式，m 代表输出的数据占的列宽，不够的在前面补上空格，n 代表小数的显示位数

例题：

1. 有如下程序段:

```
int a=3,b=15;
float f=7.5;
printf(“%.1f”,(float)a+b/2+(int)f%3);
```

执行该程序段后,运算结果为：（　　）

A）11.5　　B)11.0　　C)11　　D)12.0

三、选择结构

一）关系表达式、逻辑表达式和条件表达式

（1）C 中结果为真时值用 1 表示，结果为假时值用 0 表示；

（2）C 中的操作数是以非 0 认为是真，以 0 认为是假；

（3）数学中表示 10<a<100 的 C 语言表示形式：10<a&&a<100

（4）逻辑运算中的短路原则

例题：

1. 在 C 程序中,用(　　)表示逻辑"真"。
   A)1　B)非 0 的数　C)非 1 的数　D)大于 0 的数
2. 若有定义：intx=2,y=3,z=4;则表达式！(x+y)+z-1&&y+z%2 的值为（　　）
   A)4　B)0　C)1　D)2
3. 设 int a=0,b=0,c=0;c=++a||b++;则 a、b、c 值分别为（　）
   A）0　1　0　B）1　1　1　C）1　1　0　D）1　0　1
4. 执行语句:int a=1,b=0,c;c=a>0||++b;后,b 的值为(　　)
   A)0　B)1　C)2　D)不确定
5. 设 intx,y,z=4;x=y=++z;x=(y>z)?x+2:x++;则 x 的值是（　）
   A）4　B）5　C）6　D）7
6. 为表示关系 x>y>=z,则正确的 c 语言表达方式为（　　）。
   A)(x>y>=z)　B)(x>y)and(y>=z)
   C)(y<x)&&(y>=z)　D)(x>y)&(y>=z)
7. 若变量 ch 为 char 类型，能正确判断出 ch 为大写字母的表达式是（　）
   A）'A'<=CH<='Z'　B)(ch>='A')||(ch<='Z')
   C)('A'<=ch) and ('Z'>=ch)　D)(ch>='A')&&(ch<='Z')

运算符的优先级和结合性记忆方法

优先级

单目>双目>三目>特殊双目(赋值>逗号)

结合性

单目、三目　自右向左

双目　自左向右

特殊双目(赋值)　自右向左

二）if 语句

（1）条件

if(a)等价 if(a!=0)　if(!a) 等价 if(a= =0)

（2）else 的匹配原则：与上方最近的且没被匹配的 if 匹配

例题：

1．有以下程序

```
main( )
{   int a=0，b=0，c=0，d=0；
    if(a=1)b=1；c=2；
    else d=3；
    printf(%d,%d,%d,%d\n,a,b,c,d)；
```

}

程序输出（　　）

A)0，1，2，0　　B)0,0,0,3　　C)1,1,2,0　　D)编译有错

2．以下程序运行后的输出结果是_____

```
main()
{
    int   a=3,b=4,c=5,t=99;
    if(b<a&&a<c)     t=a;a=c;c=t;
    if(a<c&&b<c)     t=b;b=a;a=t;
    printf("%d%d%d\n",a,b,c);
}
```

三）switch 语句

（1）switch 的执行过程

（2）break 的用法

例题：

1．

```
#include<stdio.h>
    main()
      { int x=1,y=1,a=0,b=0;
        switch(x)
          { case 0:b++;
            case 1:a++;
            case 2:a++;b++;break;
            case 3:a++;b++;
          }
        printf("a=%d,b=%d\n",a,b);
      }
```

输出结果：

四）算法

1．交换：t=a;a=b;b=t

2．分段函数

输入出租车里程 S，输出应付车费 F。

S 和 F 的关系如下：

$$F=\begin{cases} 8 & S<=3 \\ 8+(S-3)*2 & S>3 \end{cases}$$

四、循环结构

一）循环条件均以条件为真（非 0）进行循环，以条件为假（0）结束循环）

例题：

1.　以下叙述正确的是（　　）

A） do-while 语句构成的循环不能用其它语句构成的循环来代替

B）do-while 语句构成的循环只能用 break 语句退出

C）do-while 语句构成的循环，在 while 后的表达式为非零时结束循环

D） do-while 语句构成的循环，在 while 后的表达式为零时结束循环

2. 设有程序段：

```
int k=10;
while(k=0)  k=k-1;
```

则循环体执行的次数为（ ）

A）10 次　　B）9 次　　C）0 次　　D）1 次

3. 下列( ) 循环不是无限循环

A) for(y=0; x=1 ; ++y); B) for(; ; x=0);

C) while (x=1) {x=1; } D) for(y=0,x=1;x> ++ y; x++)

二）三种循环的执行流程

例题：

1. 设变量 y 值为 3,执行下列循环语句后，变量 y 的值的是（ ）

```
do y++;  while(y<4);
```

A）3　B）4　C）5　D）6

2. 执行语句 for(I=1;I<4;); 后变量 I 的值是( )

A)3　B)4　C)5　D)0

三）循环嵌套的执行次数

例题：

1．

```
#include “stdio.h”
void main()
{ int I,j;
  for(I=0;I<=3;I++)
  {  for(j=0;j<I;j++)
     printf(”%d”,I);
     printf(“*\n”);
  }
}
```

输出结果：

四）break 和 continue 的用法

例题：

1．有以下程序

```
#include <stdio.h>
void main()
{  int  a=1,b;
   for(b=1;b<=10;b++)
   {  if(a>=8)   break;
      if(a%2==1)  {  a+=5;  continue;}
      a-=3;
```

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

输出结果：

2．

```
#include <stdio.h>
void main( )
{ int i, m=0, n=0, k=0;
  for(i=5;i<=7;i++)
  switch(i/6)
  { case 0:   m++;   n++;
    case 6:     n++;   break;
    default:    k++;   n++;
   }
   printf("%d,%d,%d\n", m, n, k);
}
```

输出结果：

五）算法

求和

1．求 1+2+3+……+100 之和；

2．编程输出 100 以内所有 6 的倍数及它们的和；

3．求输入的 10 个学生的成绩之和；

4．求 1-1/2+1/3-1/4+……+$(-1)^{n+1}$(1/n)

5．求 1+1/2+2/3+3/5+5/8+8/13+……（加 100 项）

1． 利用下面公式求 s 的值(求 20 项之和)

s=1/(1*2*3)-1/(2*3*4)+1/(3*4*5)-1/(4*5*6)+...+1/(19*20*21)-1/(20*21*22)

求最大、小值

1． 输入 10 个数据，求出其中的最大值；

2．输入 10 个数据，求出其中的最小值；

统计

1．输入 10 个学生的成绩，统计出及格的人数。

2．输入一行字符(以回车结束输入)，统计其中数字、字母和其它字符的个数。

画图

1． 编写程序打印以下图案（必须使用循环结构，直接输出不给分）

```
    1
    23
    456
    7890
```

2． 打印出以下图案（必须使用循环结构，直接输出不给分）

```
  *
 ***
```

*****

****

求素数

1．判断 m 是否是素数

2．编程求 100~300 之间的全部素数的和

求最大公约数

1．求两个整数 m 和 n 最大公约数。

穷举法

1．"百马百担"问题。有 100 匹马驮 100 担货，每匹大马每次驮 3 担，每匹中马每次驮 2 担，2 匹小马每次分驮 1 担。编写程序求大、中、小马的匹数(大、中、小马的匹数均不得为 0)。

2．一个素数加上 1000 以后是 43 的倍数，求满足这个条件的最小素数。

3．有 2 个小于 40 的正整数 a 和 b，a 的平方与 b 的和是 1053，b 的平方与 a 的和是 873，请编程求满足条件的 a 和 b 的值。(提示：满足条件的 a,b 的值唯一)

4．一个自然数除以 2 余 1，除以 3 余 2，除以 4 余 3，除以 5 余 4，除以 7 余 5，求满足这个条件的最小的自然数？

5．求所有的"水仙花数"之和。所谓"水仙花数"是指一个三位数，其各位数字的立方和等于该数本身。例如 153 是一个"水仙花数"，因为：153=1*1*1+5*5*5+3*3*3。

五、数组

一）数组语法

（1）数组的定义[ ]中必须是常量

（2）C 语言规定只能逐个引用数组元素而不能一次引用整个数组

（3）引用一维数组每个元素的的方法：

```
for(i=0;i<长度;i++)
    a[i];
```

（4）引用二维数组每个元素的的方法：

```
for(i=0;i<第一维长度;i++)
    for(j=0;j<第一维长度;j++)
        a[i][j];
```

（5）引用时，下标的值从 0 开始，不要超过数组的范围

（6）数组初始化时所给的数据不能超过[ ]中给定的长度，若所给的数据不足长度，其它的为 0

例题：

1. 数组定义为 int a[10][10];，则数组 a 有（　　）个数组元素。
   A）100　　B）81　　C）20　　D）121
2. 以下一维数组 a 的正确定义是（　　）
   A）int a(10);　　B)int n=10,a[n];
   C) int n;　　D)#define N 10
   Scanf(“%d”,&n);　　int a[N];
   Int a[n];

3. 设有语句”int a[]={3,4,5,9,8,7};”则表达式 a[1]-a[4]的值是（　　）

A）-6　B）6　C）-4　D）2

4. 以下不能正确定义二维数组的是（　　）

A) int a[2][2]={ {4}, {5}};B) int a[][2]={4,5,6,7};

C) int a[2][2]={ {4},5,6}; D) int a[2][]={{4,5},.{6,7}};

二）字符串

1）字符串和字符数组的区别

2）字符串函数的应用

例题：

1．以下程序的输出结果是（　）

```
#include <stdio.h>
#include<string.h>
main()
{   char st[20]=”2000\0\t\\”;
  printf(“%d,%d\n”,strlen(st),sizeof(st));
}
```

A)7,7　B)4,20　C)10,20　D)20,20

2．要将字符串 a 连接到字符串 b 后面，使用下面（　）语句

A）strcpy(a,b);　B)strcat(a,b);　C)strcpy(b,a);　D)strcat(b,a)

三）算法

一维数组常见题型:

1.数组插入

2.数组删除

3.数组逆置

4.数组查找（顺序查找、折半查找）

5．冒泡排序、选择排序

二维数组常见题型:

在矩阵上的进行操作：矩阵转置、矩阵求和、求最大值等。

字符数组常见题型：各字符串函数的实现

例题：

1．已知字符串 char　s[50]=”Iamstudent.”，请编写程序将字符 a 插入到 student 之前，结果为“Iamastudent”。

2．有一个 3×4 的矩阵，要求编程以求出其中值最大的那个元素，以及它所在的行号和列号。

程序如下：

```
#include <stdio.h>
void main()
{
 int i,j,row,col,max;
 static int a[3][4]={{3,5,1,8},{6,4,11,7},{9,3,10,2}};
```

```
max=a[0][0];
for(i=0;i<___;i++)        /*$BLANK1$*/
   for(j=0;j<4;j++)
      if(___)             /*$BLANK2$*/
         {
            max=___;       /*$BLANK3$*/
            row=i;
            col=j;
         }
printf("max=%d,row=%d,col=%d\n",max,row,col);
}
```

六、函数

一）函数语法

（1）程序是由多个函数组成的，一个程序有且只有一个 main 函数，程序的运行是从 main 函数开始到时 main 函数结束

（2）函数不能嵌套定义，可以嵌套调用

（3）使用函数的三点：定义、调用、申明（见下图）

（4）函数的参数：形参和实参

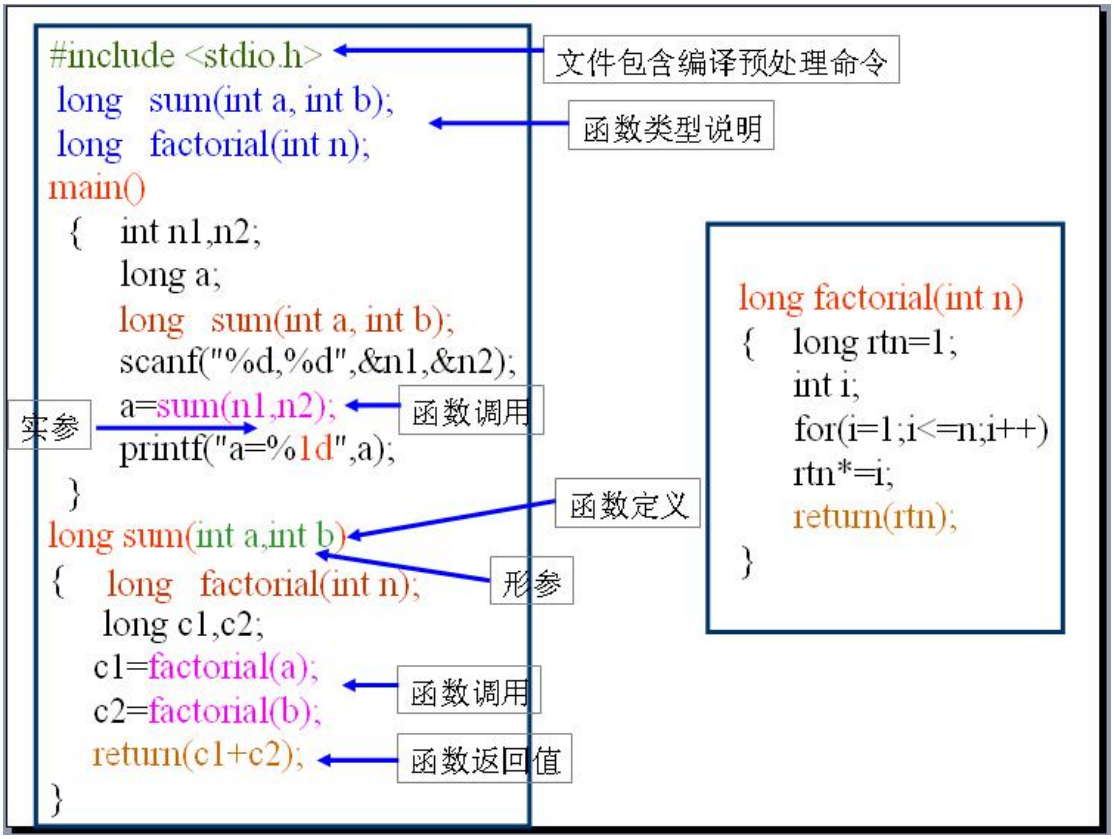
要求形参和参类型一致，个数相同

传递过程是将实参的值对应传给形参

（5）值传递和地址传递

值传递：简单类型，实参传给形参，形参发生变化不能传给实参。

地址传递：数组、指针，实参传给形参，形参发生变化也能传给实参



例题：

1. 在 C 程序中，若对函数类型未加说明，则函数的隐含类型为（　　）

   A）int　　B)double　　C)void　　D)char

2. C 语言中的函数（　）

   A）可以嵌套定义　　B）不可以嵌套调用

   C）可以嵌套定义，但不能递归调用　　D）嵌套调用和递归调用都可以。

3. 有以下函数定义：

   void　fun(int n, double　x)　{ ……}

   下面函数调用正确的是(　　)

   A)　fun(x, n)　　B) k=fun(10,12.5)

   C)　fun(int x, double y)　　D) void fun(n, x)

4. 有以下程序

```
#define　N　20
fun(int　a[],int　n,int m)
　{ int i,j;
　　for(i=m;i>=n;i--)
　　　a[i+1]=a[i];
　　}
main()
　{ int i,a[N]={1,2,3,4,5,6,7,8,9,10};
　　fun(a,2,9);
　　for(i=0;i<5;i++)
　　　printf("%d",a[i]);
　}
```

   程序运行后的输出结果是(　　)

   A）10234　　B）12344　　C）12334　　D）12234

5. 有以下程序

```
 void sum(int a[])
 { a[0] = a[-1]+a[1]; }
 main()
 { int a[10]={1,2,3,4,5,6,7,8,9,10};
   sum(&a[2]);
   printf("%d\n", a[2]);
 }
```

   程序运行后的输出结果是(　　)

   A）6　　B）7　　C）5　　D）8

6. 以下程序运行后的输出结果是：

```
 void swap(int　x,int y)
　{ int　t;
　　t=x;x=y;y=t;
　　printf("%d　　%d　　",x,y);
　}
 void swap1(int　*x,int *y)
```

```
    { int    t;
      t=*x;*x=*y;*y=*t;
      printf("%d      %d      ",*x,*y);
    }

  main()
    { int    a=3,b=4;
      swap(a,b);
      printf("%d    %d\n",a,b);
      swap1(&a,&b);
      printf("%d    %d\n",a,b);
    }
```

二）递归函数

了解递归过程（见课件）

例题：

【程序 1】

```
#include"stdio.h"
  int fun( int n)
  { if(n<10) return n;
    else
       return (n%10)*fun(n/10);
    }
 main()
    { printf("%d",fun(218));    }
```

运行结果：

【程序 2】

```
# include <stdio.h>
long fun( int n)
   { long s;
     if( n== 1|| n==2) s=2;
     else      s=n+ fun(n-1);
     return s;
    }
void    main ( )
    { printf("\n %ld", fun (5)); }
```

运行结果：

三）变量的作用域和存储类型（见课件）

例题：

1. C 语言 auto 型变量是(　　)

A) 存储在动态存储区中　B) 存储在静态存储区中
C) 存储在外存储器中　　D) 存储在计算机 CPU 的寄存器中

【程序 1】int w=2;

```
int f(int x)
   {int y=1;
    static int z=3;
    y++;z++;
    return(x+y+z);
    }
main()
    { int k;
     for(k=0;k<3;k++)
        printf("%4d",f(w++));
    }
```

运行结果：

【程序 2】以下程序运行后的输出结果是：

```
fun(int  a)
   { int b=0;static  int c=3;
    b++;    c++;
     return  (a+b+c);
   }
main()
   { int i,a=5;
    for(i=0;i<3;i++)
        printf("%d %d  ",i,fun(a));
    printf("\n");
     }
```

运行结果：

七、指针

一）指针的语法

（1）指针定义　基类型　*指针变量名；

（2）指向操作（取址操作）　指针变量=&普通变量；

（3）间接访问　*指针变量

二）指针作为参数（是地址传递，双向性）

三）指针与数组的关系

（1）与一维数组

int a[10];

定义：

数组元素类型　*指针名；

int *p;

指向数组

指针变量=数组名;

p=a

访问数组元素：

a[i]、p[i]、*(p+i)、*(a+i)

（2）与二维数组

指向二维数组行的指针变量

int a[10][4];

指针变量定义：

元素类型 (*变量名)[每行个数];

如 int (*q)[4];

联系数组

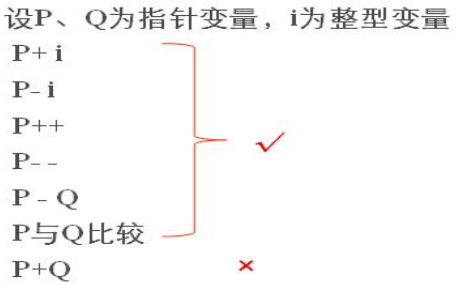
指针变量= 数组名

如 q=a;

引用任意数组元素 a[i][j]

***（ *（q+i）+j）**

（3）指向数组的指针所能进行的运算



例题：

1. 设有定义：int s[]={1,3,5,7,9},*p=&s[0];则值为 7 的表达式是（ ）

   A）*p+3 B)*p+4 C)*(p+3) D)*(p+4)

2. 若有定义及赋值：

   int a[10]={0, 1, 2, 3, 4, 5, 6, 7, 8, 9} , *p; p=a; 则引用 a 中的值是 5 的元素的表达式为（ ）

   A) a[4] B) *(p+4) C) *(a+5) D) *p[5]

3. 设 int x[]={1,2,3,4,5,6},*p=x; 则值为 3 的表达式是 （ ）

   A) p+=2, *p++ B) p+=2, * ++p

   C) p+=3 D) p+=2 , ++*p

4. 若有定义及赋值:

   int a[5][5], (*p)[5]; p=a; 则下列各式中表示地址的表达式为（ ）

   A) a[3][3] B) *(*(p+1)) C) a[2]+2 D) *(*(a+3)+1)

5. 设有一下语句

   int a[4][2]={1,2,3,4,5,6,7,8};

则*(*(a+2)+1) (　　)

A)3　B)4　C)5　D)6

四）字符串及字符串数组涵义

char　*p;：p 为字符串变量；

char　*s[10];：表示定义十个字符串，每个字符串的长度不定

例题：

1．不正确的赋值或赋初值方式是（　）

A）char str[]=”Hell0 2006”　B)char str[20];str=”Hell0 2006”;

C) char *p=”Hell0 2006”　D) char *p;p=”Hell0 2006”

2．下列语句中合法的数组定义是（　　）

A)char a[3][]={‘abc’,’12’,”fox”}

B)char a[][3]={‘abc’,’12’,”cd”}

C)char a[3][]={‘a’,”12”,”fox”}

D)char a[][3]={“a”,”12”,”cd”}

3．以下语句或语句组中，能正确进行字符串赋值的是　(　　)

A）char　*sp;　*sp="right!";　B）char s[10];　s="right!";

C）char　s[10];　*s="right!";　D）char　*sp="right!";

五）指针小结

定	含
int　i;	**定义整型变量i**
int　*p;	**p为指向整型数据的指针变量**
int　a[n];	**定义含n个元素的整型数组a**
int　*p[n];	**n个指向整型数据的指针变量组成的指针数组p**
int　(*p)[n];	**p为指向含n个元素的一维整型数组的指针变量**
int	**f为返回整型数的函数**
int *p();	**p为返回指针的函数，该指针指向一个整型数据**
int (*p)();	**p为指向函数的指针变量，该函数返回整型数**
int	**p为指针变量，它指向一个指向整型数据的指针变量**

八、宏定义、结构体与共用体

一）宏定义

宏替换过程：是原样替换，替换完成后才计算

例题：

1.　运行程序：

```
#define Tap(X)　2*X+1
main()
 { int a=6,k=2,m=1;
  a+=Tap(k+m);
```

printf(“%d\n”,a);
}则输出结果是（　　）
A）12　B）13　C）7　D）6

2. 对于以下宏定义
define　M(x)　x*x
define　N(x,y)　M(x)+M(y)
宏调用 N (2, 2+5)执行后，值为（　　）
A)　21　B)　16　C) 9　D)　19

二）结构体

（1）结构体类型、变量、数组、指针变量

```
struct　aa
{　int　x;
　float　f;
　char　c;
 }a,b[10],*p;
```

注意：struct　aa　是类型
a 是单个元素变量
b 是数组变量
p 是指针变量

不能用 aa 引用结构体中的成员

可以用 a,b,p 引用结构体中的成员，引用方法分别如下：

a.x
b[3].f;
(*p).x 或 p->x

例题：

1．设有结构体及其数组和指针变量的定义语句
struct {int x;} y[2],*p=y;
则下列表达式中不正确的表示结构体成员的是(　　)
A)(*p).x　B) (p+1).x　C)y[0].x　D)(&y[1])->x

2．设有定义 struct st { int x ; float y; } time, *t; t=&time;则对于 time 中成员 x 的正确引用是(　　)
A)　w.time.x　B)　t. x　C)　(*t). x　D)　time->x

（2）结构体所占的空间大小：所有成员字节数之和。

三）共用体

共用体所占的空间大小：所有成员字中节数最大的。

例题：

1．若定义 union ex{int I;float f;char a[10];} x;则 sizeof(x)的值是(　　)
A)4　B)6　C)10　D)16

2．设有下列结构型变量 w 的定义,则表达式”sizeof(w)”的值是（　　）

```
struct
{ long num;
  char name[15];
  union{float y;short z;} yz;
}w;
```

A)19　B)20　C)23　D)25

四）typedef

用它所定义是类型

例题：

1．以下语句中，指针 p 的 data 域正确的引用方式为（　）

```
typedef  struct node
   { int data;
     struct node lchile,rchild;
   }Btree;
Btree *p;
```

A）p.data　B)p->data　C)(*p)->data　D)p[data]

2．设有如下说明

```
typedef  struct  ST
   {  long a;
      int  b;
      char  c[2];
   } NEW;
```

则下面叙述中正确的是（　）

A）以上的说明形式非法　B）ST 是一个结构体类型

C）NEW 是一个结构体类型　D）NEW 是一个结构体变量

3．以下对结构体类型变量 td 的定义中，错误的是（　）

A）
```
typedef  struct  aa
{
   int  n;
   float  m;
 }AA;
AA  td;
```

B）
```
struct  aa
{
   int  n;
   float  m;
};
  struct  aa td;
```

C）
```
struct
{
   int  n;
   float  m;
}aa;
struct  aa td;
```

D）
```
struct
{
  int  n;
  float  m;
}td;
```

九、位运算

操作符	作用
&	位逻辑与
|	位逻辑或
^	位逻辑异或
~	位逻辑反
<<	左移
>>	右移

运算时请先转换为二进制，再按位运算

例题：请计算：

4&6=

3|5=

~4=

5>>2=

5<<2=

十、文件

文件打开方式

文件读写操作

文件关闭函数

EOF 和 feof( )

例题：

1．为了向二进制文件尾部追加内容，打开文件的方式应采用(　　)

A)　‘ab’　B)　‘rb+’　　C)　‘wb’　D)　‘wb+’

2．设有下面结构体

```
struct st
    {char name[8];
     int num;
     float s[4];
    }student[50];
```

若数组 student 中已有数据，将这些元素写到磁盘文件中，下列不正确的形式是(　　)。

A）fwrite(student,sizeof(struct st),50,fp);　　B）fwrite(student,50*sizeof(struct st),1,fp);

C）fwrite(student,25*sizeof(struct st),25,fp);

D）for(I=0;I<50;I++) fwrite(student+i,sizeof(struct st),1,fp);

3．当顺利地执行了文件关闭操作时，fclose 的返回值是（　　）

A)-1　　B）TURE　　C)0　　D)1