单项选择

**==================================================**

题号：41296

若有定义int \*p1,\*p2; 则错误的表达式是\_\_\_\_\_\_\_

A、p1+p2

B、p1-p2

C、p1<p2

D、p1=p2

答案：

A

题号：41259

以下程序的输出结果是\_\_\_\_\_\_\_\_\_\_

#include "stdio.h"

char cchar(char ch)

{ if (ch>='A' && ch<='Z') ch=ch-'A'+'a';

return ch;

}

main()

{ char s[]="ABC+abc=defDEF", \*p=s;

while(\*p)

{ \*p=cchar(\*p);

p++;

}

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

}

A、abcabcdefdef

B、abc+ABC=DEFdef

C、abcaABCDEFdef

D、abc+abc=defdef

答案：

D

题号：41289

有以下语句：

int a[5]={0,1,2,3,4,5},i; int \*p=a;

设0≤i＜5，对a数组元素不正确的引用是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A、\*(&a[i])

B、a[p-a]

C、\*(\*(a+i))

D、p[i]

答案：

C

题号：41223

C语言中规定函数的返回值的类型是由\_\_\_\_\_\_\_\_

A、return语句中的表达式类型所决定

B、调用该函数时的主调用函数类型所决定

C、调用该函数时系统临时决定

D、在定义该函数时所指定的类型所决定

答案：

D

题号：41234

以下程序的输出结果是\_\_\_\_ \_\_\_\_。

#include <stdio.h>

int n=1;

void main()

{

static int x=5;

void func();

int y;

y = n;

printf("MAIN: x=%2d y=%2d n=%2d\n",x,y,n);

func();

printf("MAIN: x=%2d y=%2d n=%2d\n",x,y,n);

func();

}

void func()

{

static int x = 4;

int y = 10;

x = x + 2;

n = n + 10;

y = y + n;

printf("FUNC:x=%2d y=%2d n=%2d\n",x,y,n); }

A、MAIN: x= 5 y= 1 n= 1

FUNC: x= 8 y=23 n=11

MAIN: x= 6 y= 1 n=11

FUNC: x= 8 y=31 n=21

B、MAIN: x= 5 y= 1 n= 1

FUNC: x= 7 y=22 n=11

MAIN: x= 5 y= 1 n=11

FUNC: x= 8 y=31 n=21

C、MAIN: x= 5 y= 1 n= 1

FUNC: x= 6 y=21 n=11

MAIN: x= 5 y= 1 n=11

FUNC: x= 8 y=31 n=21

D、MAIN: x= 6 y= 2 n= 1

FUNC: x= 6 y=21 n=11

MAIN: x= 5 y= 1 n=11

FUNC: x= 8 y=31 n=21

答案：

C

题号：41251

下面判断正确的是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A、char \*s="girl"; 等价于 char \*s; s="girl";

B、char s[4]= "boy", t[4]= "boy"; 等价于　char s[4]=t[4]= "boy"

C、char s[10]={"girl"}; 等价于 char s[10]; s[10]={"girl"};

D、char \*s="girl"; 等价于 char \*s; \*s="girl"

答案：

A

题号：41215

以下程序的输出结果是\_\_\_\_\_\_\_\_。

void fun()

{ static int a=0;

a+=2;

printf("%d",a); }

main()

{ int cc;

for(cc=1;cc<4;cc++) fun( );

printf("\n"); }

A、2222

B、2468

C、222

D、246

答案：

D

题号：41217

下列叙述中正确的是\_\_\_\_\_\_\_\_

A、C语言编译时不检查语法

B、C语言的子程序有过程和函数两种

C、C语言的函数可以嵌套定义

D、C语言所有的函数都是外部函数

答案：

D

题号：41224

阅读下面的程序：

main()

{ int swap(int,int);

int a,b;

a=3;b=10; swap(a,b);

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

swap(int a,int b)

{ int temp;

temp=a; a=b; b=temp; }

下面的说法中，正确的是\_\_\_\_\_\_\_\_

A、在main()函数中调用swap()后，能使变量a和b的值交换

B、在main()函数中输出的结果是：a=3,b=10

C、程序第2行的语句int swap();是对swap()函数进行调用

D、swap()函数的类型是void

答案：

B

题号：41272

已有变量定义和函数调用语句：int a=25; print\_value(&a);

下面函数的输出结果是\_\_\_\_\_\_\_\_\_\_

void print\_value(int \*x)

{ printf(“%d\n”,++\*x); }

A、23

B、24

C、25

D、26

答案：

D

题号：41269

以下程序段，则在main函数中使用语句\_\_\_\_\_是无意义的。

main()

{ int min();

int a, b, c, (\*p)();

p=min;

…

}

min(x, y)

int x, y;

{

…

}

A、printf("%d\n", min(a,b));

B、c=min(a, b);

C、c=(\*p)(a,b);

D、p++;

答案：

D

题号：41233

下列程序段的输出结果是\_\_\_\_ \_\_\_\_。

#include <stdio.h>

int f(int x)

{

int y;

if (x==0||x==1)

return 3;

y = x\*x-f(x-2);

return y;

}

void main()

{

printf("%d\n", f(3));

}

A、9

B、0

C、6

D、8

答案：

C

题号：41197

若用数组名作为函数调用的实参，则传递给形参的是\_\_\_\_\_\_\_\_\_\_\_\_\_

A、数组第一个元素的地址

B、数组的第一个元素的值

C、数组中所有元素的值

D、数组元素的个数

答案：

A

题号：41198

递归函数f(n)=f(n-1)+n (n>1)的递归体是\_\_\_\_\_\_\_\_\_\_\_\_\_

A、f(1)=0

B、f(0)=1

C、f(n)=f(n-1)+n

D、f(n)=n

答案：

C

题号：41196

若函数调用时参数为基本数据类型的变量，以下叙述正确的是\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A、实参与其对应的形参共享内存存储单元

B、只有当实参与其对应的形参同名时才共享内存存储单元

C、实参与对应的形参分别占用不同的内存存储单元

D、实参将数据传递给形参后，立即释放原先占用的内存存储单元

答案：

C

题号：41270

当c的值不为0时，在下列选项中能够将c的值赋给变量a、b的是\_\_\_\_\_\_\_\_\_\_\_\_

A、c=b=a;

B、(a=c)||(b=c);

C、(a=c)&&(b=c);

D、a=c=b;

答案：

C

题号：41263

已有定义　int a=2, \*p1=&a, \*p2=&a; 下面不能正确执行的赋值语句是\_\_\_\_\_\_\_\_\_\_\_\_

A、a=\*p1+\*p2;

B、p1=a;

C、p1=p2;

D、a=\*p1\*(\*p2);

答案：

B

题号：41292

以下是引用片段：

int \*f(int \*x,int \*y)

{ if(\*x<\*y) return x;

else return y; }

main()

{ int a=7,b=8,\*p,\*q,\*r;

p=&a;

q=&b;

r=f(p,q);

printf(“%d,%d,%d”,\*p,\*q,\*r); }

执行后输出结果是\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A、7,8,8

B、7,8,7

C、8,7,7

D、8,7,8

答案：

B

题号：41205

在函数调用时，以下说法正确的是\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A、函数调用后必须带回返回值

B、实际参数和形式参数可以同名

C、函数间的数据传递不可以使用全局变量

D、主调函数和被调函数总是在同一个文件里

答案：

B

题号：41214

以下函数返回a数组中最小值所在的下标，在划线处应

填入的是\_\_\_\_\_\_\_\_。

fun( int a[],int n)

{ int i,j=0,p;

p=j;

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

if(a[i]<a[p])\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

return (p); }

A、 i=p

B、a[p]=a[i]

C、p=j

D、p=i

答案：

D

题号：41298

若有定义:

int \*p,k=4; p=&k;

以下均代表地址的是\_\_\_\_\_\_\_\_\_\_\_\_\_

A、k, p

B、&k, &p

C、&k, p

D、k, \*p

答案：

C

题号：41294

void fun(int \*x, int \*y)

{ printf("%d %d", \*x, \*y);

\*x=3; \*y=4;}

main()

{ int x=1,y=2;

fun(&y,&x);

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

A、2 1 4 3

B、1 2 1 2

C、1 2 3 4

D、2 1 1 2

答案：

A

题号：41230

有以下程序

main()

{ int a,k=4,m=4,\*p1=&k,\*p2=&m;

a=p1==&m;

printf("%d\n",a); }

程序运行后的输出结果是\_\_\_\_\_\_\_\_\_\_\_\_

A、4

B、1

C、0

D、运行时出错，无定值

答案：

C

题号：41255

下面程序段的运行结果是\_\_\_\_\_\_\_\_\_\_\_\_

#include "stdio.h"

main()

{ char s[]="example!", \*t;

t=s;

while( \*t!='p')

{ printf("%c",\*t-32);

t++;

}

}

A、EXAMPLE!

B、EXAM

C、example!

D、example!

答案：

B

题号：41236

若函数的形参为一维数组，则下列说法中正确的是\_\_\_\_

A、使用函数时的对应实参必为数组名

B、形参数组可以不指定大小

C、形参数组的元素个数必须等于实参数组的元素个数

D、形参数组的元素个数必须多于实参数组的元素个数

答案：

B

题号：41240

下面的描述中错误的是\_\_\_\_\_\_

A、总是从main()函数开始执行，main函数可以调用其他函数，其他函数不能调用main函数

B、C语言程序中，其他函数不能调用main函数

C、C语言程序总是从第一个定义的函数开始执行

D、main()函数可以放在程序的开始部分

答案：

C

题号：41287

设有定义：char \*cc[2]={“1234”,”5678”};则正确的叙述是\_\_\_\_\_\_\_\_\_\_\_\_

A、cc数组的两个元素中各自存放了字符串”1234”和”5678”的首地址

B、cc数组的两个元素分别存放的是含有4个字符的一维字符数组的首地址

C、cc是指针变量，它指向含有两个数组元素的字符型一维数组

D、cc元素的值分别维”1234”和”5678”

答案：

A

题号：41264

若有说明语句：int a, b, c, \*d=&c；，则能正确从键盘读入三个整数分别赋给变量a、b、c的语句是\_\_\_\_\_\_\_\_

A、scanf("%d%d%d", &a, &b, d);

B、scanf("%d%d%d", a, b, d);

C、scanf("%d%d%d", &a, &b, &d);

D、scanf("%d%d%d", a, b,\*d);

答案：

A

题号：41303

下面程序的运行结果是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

main( ) { int a,b;

int \*p1=&a,\*p2=&b,\*t; a=10; b=20;

t=p1; p1=p2; p2=t;

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

A、10,20

B、20,10

C、10,10

D、20,20

答案：

A

题号：41212

若有以下调用语句,则正确的fun函数首部是：

main()

{ ∶

∶

int a;float x;

∶

∶

fun(x,a);

∶

∶

}

A、void fun(int m,float x)

B、void fun(float a,int x)

C、void fun(int m,float x[])

D、void fun(int x,float a)

答案：

B

题号：41204

已有以下代码

#include<stdio.h>

f( )

{ static int x = 1;

x -= 1;

printf("%d,", x);}

int main()

{ int i;

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

f();

return 0;}

则程序的输出是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A、1, 2,3,4,

B、0,1,2,3,

C、0,-1,-2,-3,

D、-1,-2,-3,-4,

答案：

C

题号：41297

以下程序段运行后，表达式\*(p+4)的值为\_\_\_\_\_\_\_\_\_\_\_\_\_ char a[]="china"; char \*p; p=a;

A、'n'

B、'a'

C、存放'n'的地址

D、存放'a'的地址

答案：

B

题号：41305

下面程序段的运行结果是\_\_\_\_\_\_\_\_\_\_\_\_\_ int a[]={1,2,3,4,5,6},\*p=a; int i,sum=0;

for(i=1;i<6;i++) sum+=\*(p++); printf("%d",sum);

A、10

B、12

C、15

D、20

答案：

C

题号：41208

下面是一个计算1至m的阶乘并依次输出的程序。程序中应填入的正确选项是\_\_\_

#include<stdio.h>

double result=1;

factorial( int j)

{ result=result\*j;

return;}

main()

{ int m,i=0,x;

printf("Please enter an integer:");

scanf("%d",&m);

for(;i++<m;)

{ x=factorial(i);

printf("%d!=%.0f\n", \_\_\_\_\_\_\_\_\_\_\_\_ ); }}

A、i,factorial(i)

B、i,x

C、j,x

D、i,result

答案：

D

题号：41242

下面是初始化指针的语句：int \*px = &a;其中指针变量的名字应该是\_\_\_\_\_\_\_\_\_\_\_

A、\*px

B、px

C、a

D、&a

答案：

B

题号：41219

请读程序:

#include <stdio.h>

char fun(char c)

{ if (c<='Z' && c>='A') c-='A'-'a';

return (c); }

main()

{ char s[81]; int k=0;

gets(s);

while(s[k])

{ s[k]=fun(s[k]); putchar(s[k]); k++;}

putchar('\n'); }

若运行时从键盘上输入 OPEN THE DOOR<CR>(<CR>表示回车),则上面程序的输出结果是\_\_\_\_\_\_\_\_。

A、oPEN tHE dOOR

B、open the door

C、OPEN THE DOOR

D、Open The Door

答案：

B

题号：41200

已有以下数组定义和f函数调用语句，则在f函数的说明中，对形参数组array的正确定义方式为\_\_\_\_\_\_。

int a[3][4];

f(a);

A、f(int array[][6])

B、f(int array[3][])

C、f(int array[][4])

D、f(int array[2][5])

答案：

C

题号：41280

若有定义：int a[5],\*p=a;则对a数组元素地址的正确引用是\_\_\_\_\_\_\_\_\_\_\_

A、p+5

B、\*a+1

C、&a+1

D、&a[0]

答案：

D

题号：41301

若有以下程序段，则叙述正确的是\_\_\_\_\_\_\_\_\_\_\_\_\_\_

char s[]="computer"; char \*p; p=s;

A、s和p完全相同

B、数组s的长度和p所指向的字符串长度相等

C、 \*p与s[0]相等

D、数组s中的内容和指针变量p中的内容相等

答案：

C

题号：41262

若有说明：int a=2, \*p=&a, \*q=p;，则以下非法的赋值语句是\_\_\_\_\_\_\_\_\_\_\_\_\_

A、p=q;

B、\*p=\*q;

C、a=\*q;

D、q=a;

答案：

D

题号：41222

在调用函数时，如果实参是简单变量，它与对应形参之间的数据传递

方式是\_\_\_\_\_\_\_\_

A、地址传递

B、单向值传递

C、由实参传给形参，再由形参传回实参

D、传递方式由用户指定

答案：

B

题号：41277

下面程序段的运行结果是\_\_\_\_\_\_\_\_

char a[ ]=”language” , \*p ;

p=a ;

while (\*p!=‟u‟)

{ printf(“%c”,\*p-32);

p++ ; }

A、LANGUAGE

B、language

C、LANG

D、langUAGE

答案：

C

题号：41276

设有程序段:char s[ ]=”china”; char \*p ; p=s ;则下面叙述正确的是\_\_\_\_\_\_\_\_\_

A、s和p完全相同

B、数组s中的内容和指针变量p中的内容相等

C、s数组长度和p所指向的字符串长度相等

D、\*p与s[0]相等

答案：

D

题号：41306

下面程序运行结果是\_\_\_\_\_\_\_\_\_\_\_\_\_\_ main( )

{ static char a[]="abcdefg",b[]="adcbehg"; char \*p=a,\*q=b; int i;

for(i=0;i<=6;i++) if(\*(p+i)==\*(q+i))

printf("%c",\*(q+i));

A、geca

B、aceg

C、bdf

D、Fdb

答案：

B

题号：41273

若有语句：int \*p,a=4;和p=&a；下面均代表地址的一组选项是\_\_\_\_\_\_\_\_\_\_\_

A、a,p,\*&a

B、&\*a,&a,\*p

C、\*&p,\*p,&a

D、&a,&\*p,p

答案：

D

题号：41254

以下正确的程序段是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A、char s[20];scanf("%s",&s[2]);

B、char \*s; scanf("%s", s);

C、char s[20];scanf("%s", &s);

D、char s[20], \*t=s; scanf("%s", t[2]);

答案：

A

题号：41209

有以下程序

#include <stdio.h>

main()

{ int x=1,y=0,a=0,b=0;

switch(x)

{ case 1:switch(y)

{ case 0:a++; break;

case 1:b++; break; }

case 2:a++; b++; break;

case 3:a++; b++; }

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

程序的运行结果是\_\_

A、a=1,b=0

B、a=2,b=2

C、a=1,b=1

D、a=2,b=1

答案：

D

题号：41295

下程序的运行结果是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

void ast (int x,int y,int \*cp,int \*dp) { \*cp=x+y; \*dp=x-y; }

main()

{ int a=4,b=3,c,d; ast(a,b,&c,&d);

printf("%d,%d\n",c,d); }

A、7,1

B、1,7

C、7,-1

D、 c、d未赋值，编译出错

答案：

A

题号：41210

以下能正确定义一维数组的选项是\_\_\_\_\_\_\_\_\_\_\_\_

A、int a[5]={0,1,2,3,4,5};

B、char a[]={'0',’1’,’2’,’3’,’4’,’5’,’\0’};

C、char a={'A’,’B’,’C’};

D、int a[5]="0123";

答案：

B

题号：41247

对函数的调用不能出现在\_\_\_\_\_\_\_\_\_\_\_

A、调用函数时传递的实参

B、引用数组元素[ ]的运算符中

C、对一个变量赋初值

D、函数的形式参数

答案：

D

题号：41201

下面程序段运行后的输出结果是\_\_\_\_\_\_\_\_\_\_\_\_（假设程序运行时输入5，3回车）

int a, b;

void swap( )

{ int t;

t=a;

a=b;

b=t; }

void main()

{ int a,b;

scanf("%d,%d",&a, &b);

swap( );

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

A、a=5,b=3

B、a=3,b=5

C、5,3

D、3,5

答案：

A

题号：41278

以下说明不正确的是\_\_\_\_\_\_\_\_\_\_\_\_\_

A、char a[10]=”china”;

B、char a[10],\*p=a; p=”china”;

C、char \*a; a=”china” ;

D、char a[10],\*p; p=a=”china”;

答案：

D

题号：41220

有以下程序

#inclde <stdio.h>

main()

{int k=4,m=1,p;

p=func(k,m); printf("%d, ",p);

p=func(k,m); printf("%d\n",p); }

func(int a,int b)

{ static int m=0,i=2;

i=m+1;

m=i+a+b;

return(m); }

输出结果是\_\_\_\_\_\_

A、6,6

B、6,12

C、6,8

D、8,8

答案：

B

题号：41282

以下与int \*q[5];等价的定义语句是\_\_\_\_\_\_\_\_\_\_\_\_

A、int q[5]

B、int \*q

C、int \*(q[5])

D、int (\*q)[5]

答案：

C

题号：41302

若有以下程序段，则不能正确访问数组元素a[1][2]的是\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int (\*p)[3];

int a[][3]={1,2,3,4,5,6,7,8,9}; p = a;

A、p[1]+2

B、p[1][2]

C、(\*(p+1))[2]

D、\*(\*(a+1)+2)

答案：

A

题号：41258

下列程序的输出结果是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

#include "stdio.h"

main()

{ int a[]={1,2,3,4,5,6,7,8,9,0}, \*p;

p=a;

printf("%d\n", \*p+9);

}

A、1

B、10

C、0

D、9

答案：

B

题号：41226

以下程序的功能是求任意两个整数a和b的最大公约数，并予以显示，程序的两划线处应分别填入\_\_\_\_\_\_\_\_。

#include <stdio.h>

#include <stdlib.h>

long cod(long n1,long n2)

{ long t;

while(n2) {\_\_\_\_\_\_\_\_; n1=n2;n2=t; }

return \_\_\_\_\_\_\_\_; }

main( )

{ long a,b,x;

printf("Please input two numbers:");

scanf("%ld%ld",&a,&b);

x=cod(a,b);

printf("%ld,%ld,%ld\n",a,b,x); }

A、t=n1和n1

B、t=n2和n2

C、t=n1%n2和n1

D、 t=n1%n2和n2

答案：

C

题号：41267

以下程序的输出结果\_\_\_\_\_\_\_\_\_

#include <stdio.h>

char \*p=”abcdefghijklmnopq”;

main()

{ while(\*p++!=’e’) ;

printf(“%c\n”, \*p);

}

A、c

B、d

C、e

D、f

答案：

D

题号：41308

设p1和p2是指向同一个字符串的指针变量，c为字符变量，则以下不能正确 执行的赋值语句是\_\_\_\_\_\_\_\_

A、c=\*p1+\*p2;

B、p2=c

C、p1=p2

D、c=\*p1\*(\*p2);

答案：

B

题号：41245

以下程序的运行结果是\_\_\_\_\_\_

int fun(int a, int b)

{ if(a>b)

return(a+b);

else

return(a-b);

}

void main()

{ int x=3, y=8, z=6, r;

r=fun(fun(x,y),2\*z);

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

}

A、-17

B、58

C、-58

D、-48

答案：

A

题号：41312

有如下语句 int a=10,b=20,\*p1,\*p2; p1=&a; p2=&b; 若要让p1也指向b，可选用的赋值语句是\_\_\_\_\_\_\_\_\_\_

A、\*p1=\*p2;

B、p1=p2;

C、p1=\*p2;

D、\*p1=p2;

答案：

B

题号：41221

下面函数的功能是\_\_\_\_\_\_\_\_。

sss(s,t)

char s[ ],t[ ];

{ int i=0;

while( t[i]) { s[i]=t[i]; i++ ; }

s[i]= '\0'; }

A、求字符串的长度

B、比较两个字符串的大小

C、将字符串s复制到字符串t中

D、将字符串t复制到字符串s中

答案：

D

题号：41228

若有语句int \*point,a=4;和point=&a;下面均代表地址的一组选项是\_\_\_\_\_.

A、a,point,\*&a

B、&\*a,&a,\*point

C、\*&point,\*point,&a

D、&a,&\*point ,point

答案：

D

题号：41250

若有说明：int \*p, a=1, b; 以下正确的程序段是\_\_\_\_\_\_\_\_

A、p=&b;

\*p=a;

B、p=&b;

scanf("%d", \*p);

C、p=&b;

scanf("%d", &p);

D、scanf("%d",&b);

\*p=b;

答案：

A

题号：41238

如果在一个函数中的复合语句中定义了一个变量，则该变量\_\_\_\_\_\_\_\_\_

A、为非法变量

B、只在该复合语句中有效

C、在该函数中有效

D、在本程序范围内有效

答案：

B

题号：41235

以下程序的输出结果是\_\_\_\_ \_\_\_\_\_。

#include <stdio.h>

int workover(int i);

int reset(int i);

int i = 0;

int main()

{

int i=5;

reset(i/2);

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

reset(i=i/2);

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

reset(i/2);

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

workover(i);

printf("i=%d", i);

}

int workover(int i)

{

i = (i % i)\*((i \* i) / (2 \* i) + 4);

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

return i;

}

int reset(int i)

{

i = i <= 2 ? 5: 0;

return i;

}

A、i=4i=3i=3i=0i=2

B、i=5i=2i=2i=0i=2

C、i=5i=4i=3i=2i=2

D、i=4i=3i=2i=0i=2

答案：

B

题号：41213

函数fun的功能是:根据以下公式计算并返回S,n通过形参传入,n的值

大于等于0。划线处应填\_\_\_\_\_\_\_\_。

1 1 1 1

S=1 - ─ + ─ - ─ + … ───

3 5 7 2n-1

float fun(int n)

{ float s=0.0,w,t,f=-1.0;

int i;

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

{f=-f;

w=f/(2\*i+1);

s+=w; }

\_\_\_\_\_\_\_\_\_\_\_\_; }

A、return (f)

B、return (S)

C、return (s)

D、return (w)

答案：

C

题号：41266

在下列结论中，只有一个是正确的，它是\_\_\_\_\_\_\_\_\_\_\_\_

A、递归函数中的形式参数是自动变量

B、递归函数中的形式参数是外部变量

C、递归函数中的形式参数是静态变量

D、递归函数中的形式参数可以根据需要自己定义存储类型

答案：

A

题号：41274

下面程序段中，for循环的执行次数是\_\_\_\_\_\_\_\_\_\_\_\_

char \*s=“\ta\018bc” ;

for ( ; \*s!=‟\0‟ ; s++)

printf(“\*”) ;

A、9

B、7

C、6

D、5

答案：

C

题号：41229

有以下程序

#include<stdio.h>

main()

{int m=1,n=2,\*p=&m,\*q=&n,\*r;

r=p;p=q;q=r;

printf("%d,%d,%d,%d\n",m,n,\*p,\*q); }

程序运行后的输出结果是\_\_\_\_\_\_\_\_\_\_\_

A、1，2，1，2

B、1，2，2，1

C、2，1，2，1

D、2，1，1，2

答案：

B

题号：41249

若有语句int \*p, a=10; p=&a; 下面均代表地址的一组选项是\_\_\_\_\_\_\_\_\_\_\_\_\_

A、&a, &\*p, p

B、\*&p, \*p, &a

C、&\*a, &a, \*p

D、a, p, \*&a

答案：

A

题号：41268

以下程序的输出结果\_\_\_\_\_\_\_\_\_

#include <stdio.h>

main()

{ static char a[ ]= "language", b[]="program";

char \*ptr1=a, \*ptr2=b;

int k;

for(k=0; k<7; k++)

if(\*(ptr1+k)==\*(ptr2+k))

printf("%c", \*(ptr1+k));

}

A、gae

B、ga

C、language

D、有语法错误

答案：

B

题号：41232

以下关于实参和形参的说法中正确的是\_\_\_\_\_\_\_

A、实参和与其对应的形参共占用一个存储单元

B、实参和与其对应的形参各占用独立的存储单元

C、只有当实参和与其对应的形参同名时才共占用存储单元

D、形参是虚拟的，不占用存储单元

答案：

B

题号：41231

函数调用语句func((exp1,exp2),(exp3,exp4,exp5));含有实参的个数为\_\_\_\_\_\_\_

A、4

B、2

C、4

D、3

答案：

B

题号：41194

在C语言中\_\_\_\_\_\_\_\_\_\_\_\_\_

A、函数的定义可以嵌套，但函数的调用不可以嵌套

B、函数的定义和调用均不可以嵌套

C、函数的定义不可以嵌套，但是函数的调用可以嵌套

D、函数的定义和调用均可以嵌套

答案：

C

题号：41290

以下程序段运行后，表达式\*(p+4)的值为\_\_\_\_\_\_\_\_\_\_\_ char a[]="china"; char \*p; p=a;

A、'n'

B、'a'

C、存放'n'的地址

D、存放'a'的地址

答案：

B

题号：41265

若有语句int a[10]={0,1,2,3,4,5,6,7,8,9},\*p=a；则\_\_\_\_\_\_\_\_不是对a数组元素的正确引用（其中0≤i<10）

A、p[i]

B、\*(\*(a+i))

C、a[p-a]

D、\*(&a[i])

答案：

B

题号：41300

若有定义int a[2][3],\*p=a;，则能表示数组元素a[1][2]地址的是

A、 \*(a[1]+2)

B、a[1][2]

C、p[5]

D、p+5

答案：

D

题号：41206

有以下程序

void ss(char \*s,char t)

{ while(\*s)

{ if(\*s==t)\*s=t-′a′+′A′;

s++; } }

main()

{ char str1[100]="abcddfefdbd",c=′d′;

ss(str1,c);

printf("%s\n",str1);}

程序运行后的输出结果是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A、ABCDDEFEDBD

B、abcDDfefDbD

C、abcAAfefAbA

D、Abcddfefdbd

答案：

B

题号：41193

在C语言中，调用一个函数时，实参变量和形参变量之间的传递是\_\_\_\_\_\_\_\_\_\_\_\_

A、地址传递

B、值传递

C、由实参传递给形参，并由形参传回给实参

D、由用户指定传递方式

答案：

D

题号：41218

以下程序的输出结果是

fun(int x,int y,int z)

{ z=x\*x+y\*y;}

main()

{ int a=31;

fun(5,2,a);

printf("%d",a); }

A、0

B、29

C、31

D、无定值

答案：

C

题号：41252

以下不能正确进行字符串赋初值的语句是\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A、char \*str="good!";

B、char str[]="good!";

C、char str[6]={'g', 'o','o', 'd','!','\0'};

D、char str[5]= "good!";

答案：

D

题号：41199

在函数调用过程中，如果函数fun A调用了函数fun B,函数fun B又调用了函数fun A，则\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A、为函数的直接递归调用

B、称为函数的间接递归调用

C、为函数的循环调用

D、C语言中不允许这样的递归调用

答案：

B

题号：41244

若有以下定义和语句，则选项中错误的语句是\_\_\_\_\_\_

int a=4,b=3,\*p,\*q,\*w;

p=&a;

q=&b;

w=q;

q=NULL;

A、\*p=a

B、w=p;

C、\*q=0;

D、\*p=\*w

答案：

C

题号：41285

下面程序段的输出是\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int a[ ]={2,4,6,8,10,12,14,16,18,20,22,24},\*q[4],k;

for (k=01; k<4; k++)

q[k]=&a[k\*3];

printf(“%d\n”,q[3][0]);

A、8

B、16

C、20

D、输出不合法

答案：

C

题号：41299

若已定义: int q=5; 对① int \*p=&q;和② p=&q; 这两条语句理解错误的是\_\_\_\_\_\_\_\_\_\_\_\_

A、①是对p定义时初始化,使p指向q;而②是将q的地址赋给p

B、①和②中的&q含义相同,都表示给指针变量赋值

C、①是对p定义时初始化,使p指向q;而②是将q的值赋给p所指向的变量

D、①和②的执行结果都是把q的地址赋给p

答案：

C

题号：41284

以下选项中，对指针变量p的正确操作是\_\_\_\_\_\_\_\_\_\_

A、int a[3], \*p;p=&a;

B、int a[5], \*p;p=a;

C、int a[5];int \*p=a=100;

D、int a[5]; int \*p1,\*p2=a; \*p1=\*p2;

答案：

B

题号：41304

以下程序段运行后变量ans的值为\_\_\_\_\_\_\_\_\_\_\_\_ int a[]={1,2,3},b[]={3,2,1}; int \*p=a,\*q=b; int k,ans=0; for(k=0;k<3;k++)

if(\*(p+k)==\*(q+k)) ans=ans+\*(p+k)\*2;

A、2

B、4

C、6

D、12

答案：

B

题号：41281

若有定义:int a[5];则a数组中首元素的地址可以表示为\_\_\_\_\_\_\_\_\_\_\_\_

A、&a

B、a+1

C、a

D、&a[1]

答案：

C

题号：41283

若有定义:int b[4][6],\*p,\*q[4];且0≤i<4,则不正确的赋值语句是\_\_\_\_\_\_\_\_\_\_

A、q[i]=b[i];

B、p=b;

C、p=b[i];

D、q[i]=&b[0][0];

答案：

D

题号：41195

两个形参中，第一个形参为指针类型、第二个形参为整型，则对函数形参的说明有错误的是\_\_\_\_\_\_\_\_\_

A、int a(float x[],int n)

B、int a(float \*x,int n)

C、int a(float x[10],int n)

D、int a(float x,int n)

答案：

D

题号：41311

以下程序中调用scanf函数给变量a输入数值的方法是错误的，其错误原因是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ void main() { int \*p,\*q,a,b; p=&a; printf("input a: "); scanf("%d",\*p); . . . }

A、 \*p表示的是指针变量p的地址

B、\*p表示的是变量a的值，而不是变量a的地址

C、\*p表示的是指针变量p的值

D、\*p只能用来说明p是一个指针变量

答案：

B

题号：41253

下面程序段的运行结果是（）。

char \*s="abcde";

s+=2;

printf("%d", s);

A、字符'c'的地址

B、字符'c'

C、cde

D、无确定的输出结果

答案：

A

题号：41260

#include "string.h"

#include "stdio.h"

main()

{ char \*p="abcde\0fghjik\0";

printf("%d\n", strlen(p));

}

程序运行后的输出结果是\_\_\_\_\_\_\_\_\_\_\_

A、15

B、6

C、5

D、12

答案：

C

题号：41211

C语言中,凡未指定存储类别的局部变量的隐含存储类别是:

A、自动(auto)

B、静态(static)

C、外部(extern)

D、寄存器(register)

答案：

A

题号：41239

#include <stdio.h>

float fun(int x, int y)

{

return ( x+y);

}

void main()

{

int a = 2,b = 5,c = 8;

printf("%3.0f\n", fun((int)fun(a+c, b), a-c));

}

A、8

B、9

C、编译出错

D、21

答案：

B

题号：41257

若已定义char s[10];则在下面表达式中不表示s[1]地址的是\_\_\_\_\_\_

A、s++

B、&s[1]

C、&s[0]+1

D、s+1

答案：

A

题号：41237

下面程序的运行结果是\_\_\_\_ \_\_\_\_。

#include <stdio.h>

char a = 'A', b = 'B';

void fun2(char a,char b)

{

printf("%3c%3c", a, b);

}

void fun1()

{

a = 'C';

b = 'D';

}

main()

{

fun1();

printf("%3c%3c", a, b);

fun2('E', 'F');

}

A、C D E F

B、C D A B

C、A B E F

D、A B C D

答案：

A

题号：41307

若有定义：int a[2][3],则对a数组的第i行j列元素地址的正确引用为\_\_\_\_\_\_\_\_\_\_\_\_

A、\*(a[i]+j)

B、(a+i)

C、\*(a+j)

D、a[i]+j

答案：

D

题号：41203

以下描述正确的是\_\_\_\_\_\_\_\_\_\_\_\_\_

A、调用函数时的实参只能是有确定值的变量

B、return()语句的括号中，可以是变量、常量或有确定值的表达式

C、C语言中，函数调用时实参和形参间的参数传递方式不都是值传递

D、实参和形参若类型不匹配，编译时将报错

答案：

B

题号：41288

已有函数max(a,b)，为了让函数指针变量p指向函数max，正确的赋值方法是\_\_\_\_\_\_\_\_\_

A、p=max;

B、p=max(a,b);

C、\*p=max;

D、\*p=max(a,b);

答案：

A

题号：41243

以下程序的输出结果是\_\_\_\_\_\_\_\_\_。

int Change(int \* pX)

{ int y=8;

y=y-\*pX;

pX=&y;

return 0;

}

int main()

{ int xx=3;

int \*pY;

pY=&xx;

Change(pY);

printf("%d", \*pY);

return 0;

}

A、3

B、不确定

C、8

D、5

答案：

A

题号：41293

以下是引用片段：

void fun(char \*c,int d)

{ \*c=\*c+1;d=d+1;

printf(“%c,%c,”,\*c,d);}

main()

{ char a=’A’,b=’a’;

fun(&b,a);

printf(“%c,%c\n”,a,b);}

程序运行后的输出结果是\_\_\_\_\_\_\_\_\_\_\_\_\_

A、B,a,B,a

B、a,B,a,B

C、A,b,A,b

D、b,B,A,b

答案：

D

题号：41216

以下程序的输出结果是\_\_\_\_\_\_\_\_。

int d=1;

fun( int p)

{ static int d=5;

d+=p;

printf("%d ",d);

return (d); }

main()

{ int a=3; printf("%d \n",fun(fun(d))); }

A、6 11 11

B、6 6 9

C、6 12 12

D、6 6 15

答案：

C

题号：41207

以下程序的正确运行结果是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

#include <stdio.h>

func(int a,int b)

{ static int m=0,i=2;

i+=m+1;

m=i+a+b;

return (m);}

main()

{ int k=4,m=1,p;

p=func(k,m);

printf("%d,",p);

p=func(k,m);

printf("%d\n",p);}

A、8,17

B、8,16

C、8,20

D、8,8

答案：

A

题号：41248

若有说明语句：int a, b, c, \*d=&c；，则能正确从键盘读入三个整数分别赋给变量a、b、c的语句是\_\_\_\_\_\_\_\_\_\_\_

A、scanf("%d%d%d", a, b, d);

B、scanf("%d%d%d", &a, &b, d);

C、scanf("%d%d%d", &a, &b, &d);

D、scanf("%d%d%d", a, b,\*d);

答案：

B

题号：41256

若有以下定义和语句：

int s[4][5], (\*ps)[5];

ps=s;

则对s数组元素的正确引用形式是\_\_\_\_\_\_\_\_\_\_\_

A、\*(ps+1)+3

B、\*(ps+3)

C、ps[0][2]

D、ps+1

答案：

C

题号：41202

下列程序的输出结果是\_\_\_\_\_\_\_\_\_\_\_

#include "stdio.h"

void fun(int a,int b,int c)

{ c=a\*b;}

int main( )

{ int c;

fun(2,3,c);

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

return 0;}

A、0

B、1

C、6

D、无法确定

答案：

D

题号：41279

若已定义char s[10]；则在下面表达式中不表示s[1]的地址是\_\_\_\_\_\_\_\_\_\_\_\_

A、s+1

B、s++

C、&s[0]+1

D、&s[1]

答案：

B

题号：41310

下面程序应能对两个整型变量的值进行交换。以下正确的说法是\_\_\_\_\_\_\_\_\_\_\_ void main() { int a=10,b=20; pirntf("(1) a=%d,b=%d\n",a,b); swap(&a,&b);

printf("(2) a=%d,b=%d\n",a,b);}

swap(int p,int q) { int t;

t=p;p=q;q=t;}

A、该程序完全正确

B、该程序有错，只要将语句swap(&a,&b); 中的参数改为a,b即可

C、该程序有错，只要将swap()函数中的形参p和q以及t均定义为指针（执行语句不变）即可

D、以上说法都不正确

答案：

D

题号：41261

下列程序段的输出结果是\_\_\_\_\_\_\_\_\_

#include "stdio.h"

void fun( int \*x, int \*y)

{ printf("%d %d ",\*x, \*y);

\*x=3;

\*y=4;

}

main()

{ int x=1, y=2;

fun(&y, &x);

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

}

A、1 2 3 4

B、2 1 3 4

C、2 1 4 3

D、2 1 1 2

答案：

C

题号：41241

C语言规定了各个函数之间调用关系，以下说法正确的是\_\_\_\_\_\_\_

A、既允许直接递归调用也允许间接递归调用

B、不允许直接递归调用也不允许间接递归调用

C、允许直接递归调用不允许间接递归调用节

D、不允许直接递归调用允许间接递归调用

答案：

A

题号：41271

变量的指针，其含义是指该变量的\_\_\_\_\_\_\_\_\_\_\_\_

A、值

B、地址

C、名

D、一个标志

答案：

B

题号：41286

下面程序的运行结果是\_\_\_\_\_\_\_\_\_\_\_

main ( )

{ int x[5]={2,4,6,8,10}, \*p, \*\*pp ;

p=x , pp = &p ;

printf(“%d”,\*(p++));

printf(“%3d”,\*\*pp); }

A、4 4

B、2 4

C、2 2

D、4 6

答案：

B

题号：41246

下面叙述中正确的是\_\_\_\_\_\_\_\_\_\_\_\_\_

A、使用全局变量有利于程序的模块化和可读性的提高

B、全局变量在定义它的文件中的任何地方都是有效的

C、全局变量在程序全部执行过程中一直占用内存单元

D、同一文件中的变量不能重名

答案：

C

题号：41227

下面的程序用来将一个十进制正整数转化成八进制数，例如输入一个正整数25，则输出31，应填入划线处的选项是\_\_\_\_\_\_\_\_。

#include <stdio.h>

main( )

{ int i=0,j=0,a,b[10 ]={0};

printf("Enter an integer:"); scanf("%d",&a);

sub(a,b);

for(;i<10;i++)

{if(\_\_\_\_\_\_\_\_) j=1;

if(j!=0) printf("%d",b[i]); } }

void sub(int c,int d[ ])

{ int e,i=9;

while(c!=0) { e=c%8; d[i]=e; \_\_\_\_\_\_\_\_\_; i--;}}

A、b[i]==0和c/=8

B、b[i]==0和c=e/8

C、b[i]!=0和c/=8

D、b[i]!=0和c=e/8

答案：

C

题号：41309

若有说明语句

char a[]="It is mine"; char \*p="It is mine";

则以下不正确的叙述是\_\_\_\_\_\_\_\_

A、a+1表示的是字符t的地址

B、p指向另外的字符串时，字符串的长度不受限制

C、p变量中存放的地址值可以改变

D、a中只能存放10个字符

答案：

D

题号：41275

下面程序段的运行结果是\_\_\_\_\_\_\_\_\_\_\_\_\_\_

char \*s=”abcde” ;

s+=2 ;

printf(“%d”,s);

A、cde

B、字符‟c‟

C、字符‟c‟的地址

D、不确定

答案：

C

题号：41291

以下程序段运行后，表达式\*(p++)的值为\_\_\_\_\_\_\_\_\_\_\_\_ char a[5]="work"; char \*p=a;

A、'w'

B、存放'w'的地址

C、'o'

D、存放'o'的地址

答案：

A

题号：41225

以下程序是将输入的一个整数反序打印出来，例如，输入1234，则输出4321；输入-1234，则输出-4321，程序中划线处应分别填入\_\_\_\_\_\_\_\_。

void printopp(long int n)

{int i=0;

if(n==0) return;

else

while(n)

{if(n>0||i==0) printf("%ld",n%10);

else printf("%ld",\_\_\_\_\_\_\_\_);

i=1;

\_\_\_\_\_\_\_\_; } }

main( )

{ long int n; scanf("%ld",&n);

printopp(n);

printf("\n"); }

A、n%10 和n=n/10

B、-n%10 和n=n/10

C、-n/10 和n=n%10

D、n%10 和n/=10

答案：

B