试题一、单选题（每小题2分，共20分）

1. sizeof(5.0) 的值为\_\_\_\_\_\_\_.。

|  |  |  |  |
| --- | --- | --- | --- |
| A．5 | B．4 | C．2 | D．8 |

1. 若变量已正确定义并赋值，表达式\_\_\_\_\_\_不符合C语言语法。

|  |  |  |  |
| --- | --- | --- | --- |
| A．a?b:c,d | B．2E1%2 | C．2>>b&c | D．a>b==c |

1. int\* p,q; int a=10; p= &a; 正确选项为\_\_\_\_\_\_\_.

|  |  |
| --- | --- |
| A．\*p++; q = p; | B．\*q++; |
| C．q = (\*p)++; | D．(\*q)++; |

1. 在 C 程序中，下面\_\_\_\_\_不能表示逻辑值“真”。

|  |  |  |  |
| --- | --- | --- | --- |
| A．1,2 | B．4>>3 | C．-0.01 | D．“false” |

1. 下面几组标识符的表示中，按照C语法均正确表示的是\_\_\_\_\_。

A. w, WORD, viod B. \_if, \_12, FILE

C. For, while, in D. a$, b1, \_a

1. 以下语句可以实现将一个文件指针fp移到文件尾的是\_\_\_\_\_\_\_.

A. rewind(fp); B.fseek(fp,0L,SEEK\_CUR); C. fseek(fp,sizeof(fp),SEEK\_SET); D. fseek(fp, 0L, SEEK\_END);

1. 若定义了 static int h[][3] = {1,2,3,4,5,6,7,8}; 那么 h[2][2]的值等于\_\_\_\_\_\_\_.

|  |  |  |  |
| --- | --- | --- | --- |
| A．0 | B．7 | C．6 | D. 8 |

1. 设变量定义为“int a; char c;”，执行语句“scanf("a=%d, c=%c",&a, &c)；”时，正确的输入是\_\_\_\_\_。

|  |  |  |  |
| --- | --- | --- | --- |
| A．10 x | B．6, c | C．a=8, c=3 | D．a=9 c=b |

1. 下列程序的输出结果是\_\_\_\_\_。

static int x, y, z;

z += (x=1) || (y=2);

printf("%d#%d", y, z);

|  |  |  |  |
| --- | --- | --- | --- |
| A．0#1 | B．2#1 | C．2#2 | D．z的值不确定 |

1. 若定义 char a[3][3]={“ad”, “ce” , “fb”}, \*s = (char \*)a; 那么下列表达式语法正确，并且其值与 a[2][1]相等的表达式是\_\_\_\_\_\_\_。

|  |  |  |  |
| --- | --- | --- | --- |
| A．\*(a+3) | B．\*(\*a+5) | C．s[2][1] | D．\*++s-2 |
|  |  |  |  |

试题二、填空题（每小题2分，共30分）

1. 变量定义如下double x=2.4,y=3.2;int a=7;，表达式x=(int)(x+y)%7/2\*a%3的值为 \_\_\_\_\_\_\_\_\_.
2. 能正确表示条件“x取值不在[1,10)和(200,210]范围内”的表达式为\_\_\_\_\_\_\_\_\_. (其中（）表示不含，[]表示含)
3. 设已定义二维数组float a[3][3], 则表达式(int)(a+1)-(int)&a[0][1]的值等于\_\_\_\_\_\_\_\_\_\_.
4. 以下程序欲计算输入整数的算术平方根，用C语言编译器编译时会存在问题，原因是\_\_\_\_\_\_\_\_\_\_.

#include <stdio.h>

int main(void)

{

int a;

scanf("%d", &a);

printf("%lf", sqrt(a));

return 0;

}

1. 输入12345#后，下列程序的输出结果是\_\_\_\_\_\_\_\_\_.

/\* '0'的ASCII值为48\*/

char c;

for(c=getchar(); (c=getchar())!='#'; c=getchar())

printf("%d ",c);

1. 以下程序段的输出是\_\_\_\_\_\_。

int f(int x)

{ return x+3.14; }

void main()

{ printf("%d", f(1.86)); }

1. 设i=10, 则putchar("0123456789ABCDEF"[i])将输出\_\_\_\_\_\_\_\_\_.
2. 执行下面程序代码后，s值为\_\_\_\_\_\_\_\_\_\_.

int i=0, j, s=0;

do

{

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

if((i\*3+j)%5 == 0) break;

i++;

if(j==3) continue;

s+=j;

} while(i<3);

1. 假设int i=0; char a='\0'; 那么执行语句while (++a) i++后，i的值是\_\_\_\_\_\_\_\_\_\_.
2. 以下程序的输出结果是\_\_\_\_\_\_\_\_\_\_.

#include <stdio.h>

void p(int \*x,int y)

{ ++ \*x;

y--;

}

void main()

{ int x=0, y=3;

p(&x, y);

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

}

1. 用typedef将一个有10个元素（元素的类型为指向整型指针的指针）的指针数组类型命名为PT，应书写为：\_\_\_\_\_\_\_\_\_\_.
2. strlen("table\ttennis")等于\_\_\_\_\_\_\_\_\_\_.
3. 从文件指针infp所指的文件中读入一个大写字母将其转换成小写后,写入文件指针outfp所指的文件中的一条语句是\_\_\_\_\_\_\_\_\_\_.
4. 以下程序段的输出是\_\_\_\_\_\_\_\_\_.

char p[3][4]={"ABC", "DEF", "GHI"};

char \*q[3],\*\*pp;

pp=q+2;

q[0]=p[1]; q[1]=p[2]; q[2]=p[0];

putchar(pp[-1][-2]+3);

1. 执行下面程序代码后，s值为\_\_\_\_\_\_\_\_\_.

int a=1, b=2, c=0, s=3;

switch(c > -a--)

{

case 1: s += b;

case 0: s += a; break;

case -1: s \*= -1;

}

试题三、程序阅读题（每小题5分，共30分）

1. 下列程序的输出是\_\_\_\_\_\_\_\_\_.

#include <stdio.h>

int s;

int f(int s)

{

static int k=0;

for(; k<=s; ++k) k++;

return s;

}

int main()

{

int s=1;

s=f(2)+f(1);

printf("%d#%d#", s, f(3));

return 0;

}

1. 程序运行时输入:aaaaa,bbbbb,ccccc,ddddd,eeee换行，程序的输出是\_\_\_\_\_\_\_\_\_.

#include <stdio.h>

int getLine(char s[],int limit);

void print(char (\*p)[10], int n);

int main()

{

char text[3][10] = {'\0'};

int i;

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

getLine(text[i], 10);

print(text, 3);

return 0;

}

int getLine(char s[ ], int limit)

{

int c, i;

for(i=0; i<limit-1 && (c=getchar())!='\n';++i)

{

s[i] = c;

}

s[i] = '\0';

return i;

}

void print(char (\*p)[10], int n)

{

int i;

for(i=0;i<n; i++) printf("%s\n", \*p++);

}

1. 文本文件sourcefile.txt内容如下：

Study

Student

下列程序的输出是\_\_\_\_\_\_\_\_\_.

int main()

{

FILE \*fp;

char buf[10];

if(( fp = fopen( "sourcefile.txt", "r" )) == NULL){

printf(" File open error!\n" );

exit(1);

}

while (!feof(fp)){

fgets(buf, 5, fp);

puts(buf);

}

return 0;

}

1. 下面程序的运行结果是\_\_\_\_\_\_\_\_\_.

#include <stdio.h>

int t(int a, int (\*f)(int))

{ return(\*f)(a\*a); }

int f(int x)

{ return 2\*x; }

int g(int x)

{ return 2+x; }

void main()

{ int x, u, v ;

x=5; u=t(x, f);v=t(x, g);

printf("%d,%d\n", u, v);

}

5. 当输入：

-1f2

-1f2

下列程序的输出结果是\_\_\_\_\_\_\_\_\_.

int f(char \*s)

{

int n=0;

while (\*s!=’\0’) {

if ((\*s>='0') && (\*s<='9'))

n = n\*16+\*s-'0';

else if ((\*s>='a') && (\*s<='f'))

n = n\*16+\*s-'a'+10;

else return -1;

s++;

}

return n;

}

int main()

{

int i,n;

char s[100];

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

scanf(“%s”, s);

if ((n=f(s+i))>=0) printf(“Yes! %d\n”, n);

else printf(“No!\n”);

}

return 0;

}

6. 下列程序的输出结果是 \_\_\_\_\_\_\_。

#include <stdio.h>

char \* f(char \*\*p)

{

int i=0, j=0;

static char s[10];

while(p[i] != NULL)

{

while(\*p[i] != '\0') p[i]++;

s[j++] = p[i][-1];

i++;

}

s[j] = '\0';

return s;

}

int main()

{

char \*ap[4]={"tin", "tango", "tip", NULL};

puts(f(ap));

return 0;

}

试题四、程序填空题（每空2分，共20分）

1. 下面的程序读取一个学生成绩的文件，并将内容输出显示在屏幕上。文件的每一行记录了学号、姓名、成绩；整个文件以-1作为结束标志。文件格式如下:

315071 张三 80

315121 李某四 90.5

315063 王二 97

-1

#include<stdio.h>

int main()

{

longnum[100];

char name[100][10];

float score[100];

int k, count = 0;

FILE \*fp = fopen("scores.txt", \_\_\_\_\_\_\_\_\_\_(1)\_\_\_\_\_\_\_\_\_\_\_\_);

if (\_\_\_\_\_\_\_\_(2)\_\_\_\_\_\_\_\_\_\_\_\_\_) return 0;

while (!feof(fp)) {

fscanf(fp, "%d", &num[count]);

if (num[count] < 0) break;

fscanf(fp, \_\_\_\_\_\_\_\_(3)\_\_\_\_\_\_\_\_\_\_\_\_);

fscanf(fp, \_\_\_\_\_\_\_\_(4)\_\_\_\_\_\_\_\_\_\_\_\_);

count++;

}

printf("There are %d students:\n", count);

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

printf("%d %s\t %.1f\n", num[k], name[k], score[k]);

\_\_\_\_\_\_\_\_\_(5)\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

return 1;

}

2. 对大于long int所能表示的正整数的运算，必须采用其它方法实现。下面的程序以字符串方式实现十进制正整数的加法运算。程序输入两个只含有'0'-'9'的字符的非空字符串（第1个字符不为'0'），表示两个超长十进制正整数，程序实现了这两个超长正整数的加法运算并以字符串返回和输出。

#include <stdio.h>

#include <string.h>

#define LEN 1000

char \* long\_long\_add(char \*a, char \*b, char \*c );

int main(void)

{

static char a[LEN+1], b[LEN+1], c[LEN+2];

printf("输入两个超长正整数(不超过1000位，'0'开始的串表示0值):\n");

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

printf("这两个正整数的和值是：\n%s \n", long\_long\_add( a, b, c ) );

return 0;

}

char \* long\_long\_add(char \*num1, char \*num2, char \*c )

{

int len1 = strlen(num1), len2 = strlen(num2);

int i = len1-1, j = len2-1, k, carry = 0; /\* carry记录进位 \*/

if( len2 > len1 ) k = len2;

else k = len1;

c[k+1] = '\0'; /\* 字符串结束标记 \*/

while( i>=0 && j>=0 ) { /\* 从个位数开始，计算到位数较少的整数 \*/

c[k] = carry + (num1[i]+num2[j] -'0'); /\* 计算当前一位数 \*/

if( c[k] > '9' ){

carry = 1; \_\_\_\_\_\_\_\_\_\_(6)\_\_\_\_\_\_\_\_\_\_\_\_;

}

else

carry = 0;

i--; j--;

\_\_\_\_\_\_\_\_\_\_(7)\_\_\_\_\_\_\_\_\_\_\_\_

}

while( i>=0 ) { /\* 如果整数num1位数较多，加算超出部分 \*/

c[k] = (carry + num1[i] > '9') ? '0' : \_\_\_\_\_\_\_\_\_\_(8)\_\_\_\_\_\_\_\_\_\_\_\_;

\_\_\_\_\_\_\_\_\_\_(9)\_\_\_\_\_\_\_\_\_\_\_\_ = (carry + num1[i] > '9') ? 1 : 0;

i--; k--;

}

while( j>=0 ) { /\* 如果整数num2位数较多，加算超出部分 \*/

c[k] = (carry + num2[j] > '9') ? '0' : (carry + num2[j]);

\_\_\_\_\_\_\_\_\_\_(9)\_\_\_\_\_\_\_\_\_\_\_\_= (carry + num2[j] > '9') ? 1 : 0;

j--; k--;

}

if(carry) \_\_\_\_\_\_\_\_\_\_(10)\_\_\_\_\_\_\_\_\_\_\_\_;

return &c[k+1];

}