**一. 单项选择题(每小题2分 共50分)**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **题 号** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **答 案** | D | A | C | D | D | D | B | C | C | A |
| **题 号** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **18** | **19** | **20** |
| **答 案** | C | D | D | A | C | A | A | A | D | C |
| **题 号** | **21** | **22** | **23** | **24** |  |  |  |  |  |  |
| **答 案** | C | C | A | A |  |  |  |  |  |  |

**二．简答题（每小题3分，共18分）**

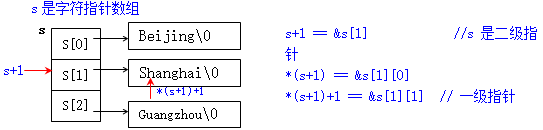
1．设整型变量x为任何一个整数，表达式 –2<x<2 的值是?

**1**

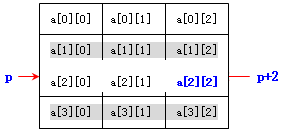
2．调用函数时，希望从提供的实参变量中得到函数的结果，则对应的形参应该是？

**指针或引用类型参数**  
　注意使用方法：   
　函数定义                    调用   
　void fun1(int x1);          int a1; fun1(a1);       //传值参数   
　void fun2(int \* x2);        int a2; fun2(&a2);      //指针参数，传地址   
　void fun3(int & x3);        int a3; fun3(a3);       //引用参数，传名

3．设有变量定义 char \*s[]={“Beijing”,”Shanghai”,”Guangzhou”}; 执行语句cout<<\*(s+1)+1; 输出是？

**hanghai**  
  s是字符指针数组   


4．设有变量定义int a[4][3],\*p=a[2]; 表达式 p+2 指向的元素是？   **a[2][2]**



5．将字符串str2拷贝到字符数组str1[]中，应调用函数？    **strcpy(str1,str2)**

6．设有函数调用语句 Count(a ,num,max,min,average); 功能是由参数max, min, average返回整型数组a前num个元素中的最大值，最小值和平均值。对应的函数原型是？

**void Count(int[], int, int &,int &, float &);**

若调用语句为  Count(a ,num, &max, &min, &average);

则函数原型为  void Count(int[], int , int\* , int\* , float \* );

**三．分析程序，写输出结果（每小题3分，共9分）**

1．求>0的元素和；>0的元素个数。   
#include<iostream.h>  
void main()  
{ int a[10]={2,4,0,-5,10,6,-8,9,6,7};  
  int i,s=0,count=0;  
  for(i=0;i<10;i++)  
    if(a[i]>0) s+=a[i],count++;  
      else continue;  
  cout<<"s="<<s<<"  count="<<count<<endl;  
}      
**s=44  count=7**

2． 测试静态变量和自动变量。静态变量第一次自动初始化为0，再调用时不改变。   
#include<iostream.h>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **a** | **b** | **a+=3** | **a+b** |
| **①** | **0** | **2** |  |  |
|  |  |  | **3** | ***5*** |
| **②** | **3** | **2** |  |  |
|  |  |  | **6** | ***8*** |
| **③** | **6** | **2** |  |  |
|  |  |  | **9** | ***11*** |
| **④** | **9** | **2** |  |  |
|  |  |  | **12** | ***14*** |

void fun();  
void main()  
{int i;  
 for(i=1;i<5;i++)  fun();  cout<<endl;   
}  
void fun()  
{static int a;  int b=2;  
 cout<<(a+=3,a+b)<<'\t';  
}  **5    8   11   14**

3．求最小公倍数   
#include<iostream.h>  
int lcm(int m, int n)  
{ int r,a,b;  
  a=m; b=n; r=m;  
  do{              //辗转相除法，求最大公约数   
  m=n; n=r; r=m%n;  
  }while(r);  
  return a\*b/n;        //求最小公倍数   
}  
void main()  
{ cout<<lcm(3,7)<<'\t'<<lcm(4,6)<<endl;  
}    
**21    12**

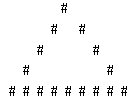
**四．根据程序功能填空。（每空2分，共10分）**

1．本程序按以下公式计算e的值，精度为1e-6。   
    http://cs.scutde.net/mnst/022/test/images/01.png  
#include<iostream.h>  
void main()  
{ double e,t,n;  
  e=0;  
t=n=\_\_\_\_\_(1)\_\_\_\_\_;       **1.0**  
  while(\_\_\_\_\_(2)\_\_\_\_\_)     **t>=1e-6**  
  { e+=t;  
    \_\_\_\_\_(3)\_\_\_\_\_;        **t=t/n**  
    n=n+1.0;  
  }  
 cout<<"e="<<e<<endl;  
}

2．本程序由主函数输入一字符串，调用函数，把该字符串中的数字0～9转换成小写字母a～j；所有小写字母转换成大写字符。然后在主函数输出转换后的字符串。   
#include<iostream.h>  
#include<ctype.h>  
\_\_\_\_\_(4)\_\_\_\_\_          **void change(char\*, char\*);**   
void main()  
{char str1[20], str2[20];  
 cin>>str1;  
 change(str1,str2);  
 cout<<str2<<endl;  
}  
void change(char \*s1, char \*s2)  
{ while(\_\_\_\_\_(5)\_\_\_\_\_)     **\*s1**  
 {if(\*s1>='0'&&\*s1<='9')  
     \*s2=\*s1 + 'a' -' 0'   
   else \*s2=toupper(\*s1);  
  s1++ ; s2++;  
 }  
 \*s2='\0';  
}

**五．程序设计**

1. 编写函数输出以下形状的图形，其中构成图形的符号和输出的行数通过参数传送。(6分)



**答案：**  
#include<iostream.h>  
#include<iomanip.h>  
void print\_figure(int row, char tag)    　//行参数和符号参数   
{int i;  
 cout<<setw(row-1)<<" "<<tag<<endl;     //输出第1行   
 for(i=2;i<row;i++)             //输出第2行到row-1行   
   cout<<setw(row-i)<<" "<<tag<<setw(2\*(i-1)-1)<<" "<<tag<<endl;  
 for(i=1;i<=2\*row-1;i++)cout<<tag;      //输出最后1行   
 cout<<endl;  
}  
void main() { print\_figure(5,'\*');}

２．编写一个程序，实现如下功能：(9分)

（1）输入k（<100）个整数到数组x[100]中；   
（2）计算k个数的平均值及大于平均值的元素个数。   
**答案：**  
#include<iostream.h>  
void main()  
{int x[100],k,i,n;  
 double sum=0.0,ave;  
 cout<<"How many Data ?\n";  
 cin>>k;  
 for(i=0;i<k;i++)       //求和   
 { cin>>x[i]; sum+=x[i];}  
 ave=sum/k;             //求平均值   
 n=0;  
 for(i=0;i<k;i++)       //求大于平均值的元素个数   
   if(x[i]>ave) n++;  
 cout<<"average="<<ave<<"\n";  
 cout<<"There are "<<n<<" elements large than average.\n";  
}