选择题

（指针）0、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？

1. ps+1
2. \*(ps+3)
3. ps[0][2]
4. \*(ps+1)+3

（指针）1、有如下代码段：

#include<stdio.h>

#include<stdlib.h>

int a=0;

int \*p;

int main(){

float b;

double \*dp=(double\*)malloc(sizeof(double))

}

问在内存中存储在同一存储区的变量为？

1. a,\*dp
2. a,b
3. b,dp
4. p,dp

（指针）、若有说明：int a=2,\*p=&a,\*q=p; 则下列赋值语句非法的是？

A) p=q

B) \*p=\*q

C) a=\*q

D) q=a

（指针）3、若有说明：int a[]={1,3},\*p=a;则printf("%d",\*a++);的输出结果为？

A) a的地址++

B) 1

C) 2

D) 3

（指针）4、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

（指针）5、若有语句：int \*p,a=10;p=&a; 下列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

（指针）6、以下程序调用scanf函数给变量a输入数值的方法是错误的，其错误原因是？

#include<stdio.h>

int main(){

int \*p,\*q,a,b;

p=&a;

printf("input a:");

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

...

}

1. \*p表示的是指针变量p的地址
2. \*p表示的是变量a的值，而不是变量a的地址
3. \*p表示的是指针变量p的值
4. \*p只能用来说明p是一个变量指针

（指针）7、下列判断正确的是？

A) char \*s="zjgsu"; 等价于 char \*s;\*s="zjgsu"

B) char s[10]={"zjgsu"}; 等价于 char s[10];s[10]={"zjgsu"};

C) char \*s="zjgsu"; 等价于 char \*s;s="zjgsu"

D) char s[10]="zjgsu",t[10]="zjgsu"; 等价于 char s[10]=t[10]="zjgsu";

（指针）8、设 char=\*s="zj\020gsu\n"; 则指针变量s指向的字符串所占的字节数是？

A) 3

B) 11

C) 10

D) 8

（指针）9、有如下程序段：

char \*s="zj\020gsu\n";

for(;\*s!='\0';s++){

printf("\*")

}

则该程序的输出结果为？

1. \*\*
2. \*\*\*\*\*\*\*\*\*\*\*
3. \*\*\*\*\*\*\*\*\*\*
4. \*\*\*\*\*\*\*\*

（指针）10、下列赋值语句无法赋值成功的是？

A) char str[5]="zjgsu";

B) char \*str="zjgsu";

C) char str[]="zjgsu";

D) char str[5]={'S','C','I','E'};

（指针）11、若指针p已正确定义，要使p指向两个连续的整型动态存储单元，不正确的是？

A) p=2\*(int\*)malloc(sizeof(int));

B) p=(int\*)malloc(2\*sizeof(int));

C) p=(int\*)malloc(8);

D) p=(int\*)calloc(2,sizeof(int));

（指针）12、若已定义char s[10]; 则在下面表达式中不表示s[1]地址的是？

1. s+1
2. S++
3. &s[0]+1
4. &s[1]

（指针）13、执行以下程序后，a的值为？

#include<stdio.h>

int main(){

int a,k=4,m=6,\*p=&k,\*q=&m;

a=p==&m;

}

1. -1
2. 1
3. 0
4. 4

（指针）14、执行以下程序后，b的值为？

#include<stdio.h>

int main(){

int b,k=4,m=6,\*p=&k,\*q=&m;

b=(-\*p)/(\*q)+7;

}

1. 5
2. 6
3. 7
4. 10

（指针）15、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,\*q,\*p);

}

程序运行后的输出结果是？

1. 1,2,1,2
2. 1,2,2,1
3. 2,1,2,1
4. 2,1,1,2

（指针）16、有以下程序：

#include<stdio.h>

int main(){

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

int \*p1=&a,\*p2=&b,\*p=&c;

\*p=\*p1\*\*p2;

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

}

程序运行后的输出结果是？

1. 1
2. 2
3. 3
4. 4

（指针）17、在16位系统上，若有定义int a[]=(10,20,30},\*p=&a;当执行p++; 后，下列说法错误的是？

1. p向高地址移动了一个字节
2. p向高地址移动了一个存储单元
3. p向高地址移动了两个字节
4. p与a+1等价

（指针）18、有以下程序段：

int a[10]={1,2,3,4,5,,6,7,8,9,10},\*p=&a[2],b;b=p[4];

b的值为？

1. 4
2. 5
3. 6
4. 7

（指针）19、有以下程序：

void fun(char \*c,int d){

\*c=\*c+1;

d=d+1;

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

}

void 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？

1. B,a,B,a
2. a,B,a,B
3. A,b,A,b
4. b,B,A,b

（指针）20、若有以下程序：

void main(){

char \*a[3]={"I","love","zjgsu"};

char \*\*ptr=a;

printf("%c %s",\*(\*(a+2)+1),\*(ptr+1));

}

这段程序的输出是？

1. I l
2. o o
3. o zjgsu
4. j love

（指针）21、有如下代码段：

char \*c[]={"HELLO","NEW","WORLD","SAYHI"};

char \*\*cp[]={c+3,c+2,c+1,c};

char \*\*\*cpp=cp;

int main(){

printf("%s",\*\*++cpp);

printf("%s",\*--\*++cpp+3);

printf("%s",\*cpp[-2]+3);

printf("%s",cpp[-1][-1]+1);

}

则该程序的输出结果为？

1. WORLD,LO,SAYHI,EW
2. WORLD,LO,HI,NEW
3. NEW,LO,SAYHI,EW
4. WORLD,LO,HI,EW

（指针）22、有如下定义：char \*s="asdas\0accnov\0"; 则printf("%d",strlen(s));的输出为？

1. 5
2. 6
3. 15
4. 13

（指针）23、若有说明：int i,j=7,\*p=&i;则与i=j;等价的语句是？

1. i=\*p;
2. \*p=\*&j;
3. i=&j;
4. i=\*\*p;

（指针）24、int j,\*p=&j;要实现j的内容加10，错误的是？

1. \*p=\*&j+10
2. j=\*p+10
3. \*p=\*p+10
4. \*p=&j+10

（指针）25、若有以下定义char s[]={"12345"},\*p=s;则下列表达式中错误的是？

1. \*(p+1)
2. \*(s+2)
3. p="ABC"
4. s="ABC"

（指针）26、若有如下定义和语句，则输出结果是？

int \*\*p,\*p,a=10,b=20;

pp=&p;p=&a;p=&b;

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

1. 10,20
2. 10,10
3. 20,10
4. 20,20

（指针）27、设有定义语句：int (\*ptr)[10]; 其中的ptr是？

1. 10个指向整型变量的指针
2. 指向10个整型变量的函数指针
3. 一个指向具有10个元素的一维数组的指针
4. 具有10个指针元素的一维数组

（指针）28、以下程序的输出结果为？

void prt(int \*x,int \*y,int \*z){

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

}

int main(){

int a=10,c=20,b=40;

prt(&a,&b,&c);

prt(&a,&b,&c);

}

1. 11,42,31 12,22,41
2. 11,41,20 12,42,20
3. 11,21,40 11,21,41
4. 11,21,40 12,22,40

（指针）29、若有以下定义语句：

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

则值为3的表达式是？

1. p+=2,\*(++p)
2. p+=2,\*p++
3. p+=3,\*p--
4. p+=2,++\*p

（指针）30、程序中对fun函数有如下声明：

void \*fun();

此声明的含义是？

1. fun函数无返回值
2. fun函数的返回值可以是任意的数据类型
3. fun函数的返回值是无值型的指针类型
4. 指针fun()指向一个函数

填空题

（指针）1、设有以下定义和语句：

int a[3][2]={10,20,30,40,50,60},(\*p)[2];

p=a;

则\*(\*(p+2)+1)的值为

60

（指针）2、若有如下说明和语句，则在执行for语句后，\*(\*(pt+1)+2)表示的数组元素是

int t[3][3]={1,2,3,4,5,6,7,8,9},\*pt[3],k;

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

pt[k]=&t[k][0];

t[1][2]

（指针）3、若有以下定义：

int a[]={1,3,5,7,9},\*p;

则下列程序段的输出结果为

p=a;

\*(p+2)+=2;

printf("%d,%d\n",\*\*p,\*(p+2));

6,10

（指针）4、有以下程序：

#include<stdio.h>

#include<string.h>

int main(){

char b1[8]="abcdefg",b2[8],\*pb=b1+3;

while(--pb>=b1){

strcpy(b2,pb);

}

printf("%d\n",strlen(b2));

}

输出结果是

7

（指针）5、有如下程序：

#include<stdio.h>

int main(){

int x=1,y=2;

int func(int \*a,int \*b);

y=func(&x,&y);

x=func(&x,&y);

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

return 0;

}

int func(int \*a,int \*b){

if(\*a>\*b){

\*a-=\*b;

}

else{

\*a--;

}

return (\*a+\*b);

}

该程序的输出结果为

8,4

（指针）6、有如下程序：

#include<stdio.h>

int main(){

int s[6][6],i,j;

for(i=0;i<6;i++){

for(j=0;j<6;j++){

\*(\*(s+i)+j)=i-j;

}

}

for(j=0;j<6;j++){

for(i=0;i<6;i++)[

printf("%4d",\*(\*(s+i)+j));

}

printf("\n");

}

return 0;

}

则该程序输出的第三行内容为

-2 -1 0 1 2 3

（指针）7、指针除了可以指向变量，也可以指向

函数

（指针）8、有如下程序段：int a[]={1,3,5},\*p=a; 则++\*p++结果为

2

（指针）9、有如下程序：

void swap(int \*a,int b){

int t=b;

b=\*a;

\*a=t;

}

int main(){

int a=5,b=10;

swap(&a,b);

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

}

则输出结果为

10,10

（指针）10、当对指针p动态申请内存空间后，在使用的过程中，发现之前申请的内存不够，可使用 函数申请更大的空间。

realloc

（结构与联合）1、设如下结构体定义：

struct student{

int num;

char name[20];

int sex;

}stu,\*p=&stu;

使用指针变量p对stu中的num赋值1701的方法是

p->num=1701;

(\*p).num=1701;

（结构与联合）2、设有下列结构定义语句：

struct score{

int Chinese;

int math;

}scores[2]={{80,85},{89,93}},\*p=scores;

则++p->math的值为 ,++(p->math)的值为

86,86

（结构与联合）3、设有如下枚举类型定义：

enum color{red,black=3,green,yellow};

则printf("%d\n",green);输出的值为

4

（结构与联合）4、设某结构体定义如下：

struct INT{

int n;

};

则sizeof(struct INT)\_\_\_\_\_sizeof(int)（填“>”“<”或“=”）？

>

（结构与联合）5、已知有如下定义：

#define INT int

请用typedef改写上述代码以实现相同功能

typedef int INT;

（结构与联合）6、有如下结构体定义：

struct student{

long long num;

char name[20];

};

则该结构体的成员占用字节数为

28

（结构与联合）7、有如下结构体定义：

struct student{

int num;

char name[20];

union{

int math;

char PE;

}sele;

}st;

则与&st.math地址相同的有 和

&st,&st.PE

（结构与联合）8、有如下结构体定义：

struct student{

int num;

char name[20];

union{

int math;

int PE;

}sele;

}st;

st.sele.math=90;

则语句printf("%d\n",st.sele.PE);的输出结果是

90

（结构与联合）9、有如下结构体定义：

struct student{

int num;

char name[20];

}

要使该结构体能构成一个链表，还需在结构体里增加新的成员（命名为next），所增加的语句为

struct student \*next;

struct student\*next;

struct student\* next;

（结构与联合）10、有如下结构体定义：

struct student{

int num;

char name[20];

union{

int Chinese;

char PE;

};

};

则上述结构体成员所占用的内存为 字节

28

判断题

（指针）1、为避免内存泄漏，使用malloc函数动态申请的内存空间在使用完之后应及时调用free()函数释放内存空间。

√

（指针）2、C库中有标准的malloc和free函数，它们的使用不受分配、释放之间顺序的限制。

√

（指针）3、函数指针和变量指针一样，可以进行诸如fp±i,fp1-fp2,fp++等运算。

×

（指针）4、已知一级指针需要4个字节的存储空间，则二级指针需要8个字节存储空间。

×

（指针）5、已知int \*p;则p++使p向后移动了一个字节。

×

程序填空题

（指针）1、以下程序是使用Eratosthenes实现的求n以内正整数的素数的筛法，请阅读程序并填空。

#include<stdio.h>

#include<stdlib.h>

struct num {

int x;

int flag;

struct num \*next;//①

};

void Eratosthenes(int n)

{

int i, count = 0;

struct num \*head = NULL, \*p, \*q;

for (i = n; i > 1; i--)

{

p = (struct num \*)malloc(sizeof(struct num));//②

p->x = i;

p->flag = 1;

p->next = head;

head = p;//③

}

for (q = head; q != NULL; q = q->next)

if (q->flag == 0)

continue;

else

for (p = q->next; p != NULL; p = p->next)//④

if (p->x % q->x == 0)

p->flag = 0;

for (p = head; p != NULL; p = p->next)

if (p->flag == 1)//⑤

{

printf("%d ", p->x);

count++;

if (count == 10)

{

count = 0;

printf("\n");

}

}

}

int main()

{

int n;

scanf("%d", &n);

Eratosthenes(n);

}

（指针）2、在以下程序中，输入一个字符串（字符串长度不超过100），将会统计该字符串中的字母、数字、空格和其他字符的数目，请阅读程序并填空。

#include<stdio.h>

#include<stdlib.h>//①

int main(){

char \*s;

int ch = 0, num = 0, blank = 0,oth = 0, i;

s=(char\*)malloc(sizeof(char)\*105);//②

gets(s);//③

for(i=0;\*(s+i)!=0;i++){//④

if((\*(s+i)>='a'&&\*(s+i)<='z')||(\*(s+i)>='A'&&\*(s+i)<='Z')){//⑤

ch++;

}

else if(\*(s+i)>='0'&&\*(s+i)<='9'){

num++;

}

else if(\*(s+i)==' '){

blank++;//⑥

}

else{

oth++;

}

}

printf("character : %d\nnumber : %d\nblank : %d\nother : %d\n", ch, num, blank, oth);

}

（指针）3、以下程序用指针解决括弧匹配问题（字符串只包含小括号，且长度不超过100），请阅读程序并填空。

#include<stdio.h>  
#include<stdlib.h>

int main(){

char \*s, \*match;

int rear=0, i, flag=1;

s = (char\*)malloc(sizeof(char)\*105);

match = (char\*)malloc(sizeof(char)\*105);

gets(s);//①

for(i=0;\*(s+i);i++){

if(\*(s+i)=='('){

\*(match+rear++)='(';//②

}

else if(rear>0){//③

rear--;

}

else{

flag=0;

break;//④

}

}

if(flag==1&&rear>0){//⑤

flag=0;

}

printf("%s\n",flag?"match succeed":"match failed");

}

（指针）4、下面程序计算杨辉三角并存于二级指针指向的存储空间，请阅读程序并填空。（要求：二级指针分配的内存恰好存储所需的数，不得额外申请）

#include<stdio.h>

#include<stdlib.h>

int main(){

int n;

scanf("%d",&n);//n>2

int \*\*tri;

tri=(int\*\*)malloc(sizeof(int\*)\*n);

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

tri[i]=(int\*)malloc(sizeof(int)\*(i+1));//①

}

tri[0][0]=tri[1][0]=tri[1][1]=1;

for(int i=2;i<n;i++){

tri[i][0]=tri[i][i]=1;//②

for(int j=1;j<i;j++){

tri[i][j]=tri[i-1][j-1]+tri[i-1][j];//③

}

}

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

for(int j=0;j<=i;j++){

printf("%d%c",tri[i][j]," \n"[j==i]);

}

}

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

free(tri[i]);

}

free(tri);//④

}

（指针）5、以下程序用于实现删除字符串中指定字符，请阅读程序并填空。

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

int main(){

char \*s=(char\*)malloc(sizeof(char)\*100),c;

scanf("%s %c",s,&c);

int step=0,now=0;

for(int i=0;i<strlen(s);i++){

if(s[i]!=c){

s[step++]=s[i];

}

}

s[step]=0;

printf("%s",s);

}

程序改错题

（指针）1、下面程序实现对一个n×n的矩阵进行转置，请阅读程序并改错。

#include<stdio.h>

#include<stdlib.h>

void swap(int \*a,int \*b){

int t=a;a=b;b=t;//int t=\*a;\*a=\*b;\*b=t;

}

int main(){

int n,\*\*matrix;

scanf("%d",&n);

matrix=(int\*\*)malloc(sizeof(int));//matrix=(int\*\*)malloc(sizeof(int\*)\*n);

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

matrix[i]=(int\*)malloc(sizeof(int)\*n);

}

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

for(int j=0;j<n;j++){

scanf("%d",&matrix[i][j]);

}

}

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

for(int j=i+1;j<n;j++){

swap(matrix[i][j],matrix[j][i]);//swap(&matrix[i][j],&matrix[j][i]);

}

}

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

for(int j=0;j<n;j++){

printf("%d ",matrix[i][j]);

}

printf("\n");

}

}

（指针）2、下面程序实现一个函数“itoa(int i,char \*s)”，把一个整型数i转换成字符串放到指针s指向的字符数组中。请阅读程序并改错。

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

void reverse(char \*s){

int len=strlen(s);

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

char c=s[i];s[i]=s[len-i];s[len-i]=c;//char c=s[i];s[i]=s[len-i-1];s[len-i-1]=c;

}

}

void my\_itoa(int i,char \*s){

if(!i){

return;

}

\*s=i%10;//\*s=i%10+'0';

\*(s+1)=0;

my\_itoa(i/10,s+1);

}

int main(){

char \*s=(char\*)malloc(sizeof(char));//char c=s[i];s[i]=s[len-i-1];s[len-i-1]=c;

int n;

scanf("%d",&n);

my\_itoa(n,s);

reverse(s);

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

}

（指针）3、以下函数实现把字符串s2的前n个添加到字符串s1的尾部。请阅读程序并改错。

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

void strcat\_n(char \*s1,char \*s2,int n){

int len=strlen(s1);

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

if(!s2[i]){

break;

}

s1[len++]=s2[i];

s1[len++]=0;//s1[len]=0;

}

}

int main(){

char \*s1,\*s2;//char \*s1=(char\*)malloc(sizeof(char)\*10),\*s2=(char\*)malloc(sizeof(char)\*10);

int n;

scanf("%s%s%d",&s1,&s2,&n);//scanf("%s%s%d",s1,s2,&n);

strcat\_n(s1,s2,n);

printf("%s\n",s1);

}

（指针）4、以下函数实现比较两个字符串是否相等。请阅读程序并改错。

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

int my\_strcmp(char \*s1,char \*s2){

int len=strlen(s1)>strlen(s2)?strlen(s2):strlen(s1);

for(int i=0;i<len;i++){//i<len+1

if(s1[i]!=s2[i]){//(!s1[i])||(!s2[i])||s1[i]!=s2[i]

return s1[i]-s2[i];

}

}

}

int main(){

char \*s1=(char\*)malloc(sizeof(char)\*10),\*s2=(char\*)malloc(sizeof(char)\*10);

scanf("%s%s",s1,s2);

printf("%s\n",my\_strcmp(s1,s2)?"Equal":"Not equal");//"Not equal":"Equal"

}

（指针）5、以下函数实现将字符串s1从第m个字符开始的所有字符复制成另一个字符串。

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

void strcpy\_m(char \*s1,char \*s2,int m){

int len=strlen(s1),now=0;

if(len<m){

s2=0;

return;

}

for(int i=m;i<len;i++){//for(int i=m-1;i<len;i++){

s2[now++]=s1[i];

s2[now]=0;

}

}

int main(){

char \*s1=(char\*)malloc(sizeof(char)\*10),\*s2=(char\*)malloc(sizeof(char)\*10);

int m;

scanf("%s%s",s1,&m);//scanf("%s%d",s1,&m);

strcpy\_m(s1,s2,m);

printf("%s\n",s2);

}

编程题

（指针）1、

问题描述

编写一段代码，实现从n个实数型的数据中求出最大值和次大值。

输入

输入第一行为一个整数T，表示接下来有T组数据，对于每组数据：第一行为一个整数n，接下来一行包含n个实数。

输出

对于每组数据，输出包含两个实数，即为每组实数的最大值和次大值。

样例输入

1

10

1 2 3 4 5 6 7 8 9 10

样例输出

10 9

（指针）2、

问题描述

有n个人围成一圈，顺序排好（从1开始编号）。从第1个人开始报数（从1-m报数），凡报到m的人退出圈子，问最后留下来的是原来第几号的那位？

输入

输入第一行为一个整数T，表示接下来有T组数据，对于每组数据：第一行有两个整数n，m。

输出

输出最后留下来那个人的初始编号。

样例输入

1

4 6

样例输出

3

（指针）3、

问题描述

编写程序，实现如下功能：输入月份号，输出该月的英文月名。

输入

输入为一个整数。

输出

输出该月的英文月名。

样例输入

5

样例输出

May

（指针）4、

问题描述

输入一个字符串，内有数字和非数字字符。将其中连续的数字作为一个整数，依次放到数组中。

输入

输入有一个整数T，表示T组测试数据。对于每组测试数据，有一个字符串s（所有s的总长度不超过100000）。

输出

输出包括一个整数n，表示该字符串中整数的个数，接下来有n行，依次输出这n个整数。

样例输入

1

A123X456\_17960?302tab5876

样例输出

5

123

456

17960

302

5876

（指针）5、

问题描述

请编程实现一个简易的学生管理系统，实现班级的成绩处理功能。

输入

输入有一个整数T，表示有T组测试数据。对于每组测试数据：第一行是两个整数n，m，表示有n个学生，m门课程，接下来有n行，每一行包括学生m门课的成绩（m个整数）（保证所有学生每门课都存在成绩）。

输出

输出为1行，包含m个数（保留小数点后两位），依次表示各门课的平均成绩。

样例输入

1

2 1

90

95

样例输出

92.50

（指针）6、

问题描述

有一个m×n的矩阵，求出其中的最大值及最大值的位置坐标（坐标从0开始）。

输入

输入第一行包括一个T，表示T组测试样例，每组数据第一行有两个数m，n，表示m×n的矩阵。接下来有m行，每行n个整数，表示矩阵中的值。

输出

对于每组测试样例输出一行，包含三个数字，分别为最大值，行坐标，列坐标。

样例输入

1

2 2

1 2

3 4

样例输出

4 1 1

（指针）7、

问题描述

编写一个程序，实现对一个字符串（不包含空格）的前n的字符用指定字符c替代（如果字符串长度不足n，则把这个字符串都变为指定字符，但是长度不变）。

输入

输入包含一个整数T，表示T组数据。每组数据包含一个字符串s，一个字符c，一个整数n。

输出

输出一个被替代后的字符串。

样例输入

1

asdfghjkl \* 3

样例输出

\*\*\*fghjkl

（指针）8、

问题描述

编写一个程序，对字符串进行操作，把字符串前n个字符移动到字符串末尾（如果字符串长度不足n，则不变）

输入

输入包含一个整数T，表示T组数据。每组数据包含一个字符串s和一个整数n。

输出

输出移动后的字符串。

样例输入

1

asdfghjkl 3

样例输出

fghjklasd

（指针）9、

问题描述

编写一段程序，实现将整数按输入顺序的逆序输出。

输入

输入包含一个整数T，表示T组数据。每组数据第一行包含一个整数n，表示n个整数，第二行包含n个整数，以空格隔开。

输出

逆序输出所输入的n个整数。

样例输入

1

10

1 2 3 4 5 6 7 8 9 0

样例输出

0 9 8 7 6 5 4 3 2 1

（指针）10、

问题描述

利用指向行的指针变量求一个m×n的矩阵的各行元素和。

输入

输入包含一个整数T，表示T组数据。每组数据第一行为两个整数m，n，表示m×n的矩阵。接下来有m行，每行n个整数，表示矩阵中的值。

输出

输出一行包含m个整数，表示各行元素之和。

样例输入

1

2 2

1 2

3 4

样例输出

3 7