

# 暨南大学C语言程序设计实验作业合订本

2022年春

## 1. 实验一



### 实验一 Hello World



1. 在线安装Visual studio Community 2019
2. 新建一个工程
3. 输出“Hello World!”

```
#include<stdio.h>
int main(){
    printf("hello world");
    return 0;
}
```

- 每个程序员的必备小仪式

## 2. 实验二



## 实验二 尺寸转换



### ➤ 实验内容

- a) 用户输入英制尺寸。
- b) 输出对应多少米。
  - e.g., 英制尺寸5尺7 转换为  $(5 + 7 / 12) \times 0.3048 = 1.7018$  米

### ➤ 注意事项

1. 本实验可能会遇到预料之外的错误，请通过IDE的调试功能查找问题，并分析原因；
2. 独立完成。

```
#include<stdio.h>
int main(){
    int inch,cun;
    float meter;
    scanf("%d,%d",&inch,&cun);
    meter=inch*0.3048+cun*0.0254;
    printf("The meter is %f\n",meter);
}
```

- 要注意的是，在C语言中两个整型变量相除，其结果是结果向下取整，所以题目中的转换公式我们需要将 cun 的除数 12 先同 0.3048 相乘,再与inch计算的结果相加。
- 输出的时候要注意 printf 函数的数据类型要统一

### 3. 实验三



## 实验三 字符编码



### ➤ 实验内容

将用户输入的字符编码，规则如下：对于字符`char_x`，假设`char_x`是英文字母表中的第`n`个字母，那么用`char_x`后面第`n`个字母代替该字符。例如，用户输入`b`，由于`b`是英文字母表中的第2个字母，按照编码规则，用`b`后面的第2个字母替换`b`，因此输出`d`。

### ➤ 注意事项

1. 本实验约定用户只输入英文字母；
2. 英文字母表形成一个环，例如`z`是第26个字母，那么在此环中`z`的下一个字母为`a`，`y`后面的第3个字母为`b`，依次类推；
3. 允许用户输入大小写。

### ➤ DDL

1. 下课前。

```

#include<stdio.h>
int main(){
    char char_x, m, n, a, A;
    a = 97; A = 65;
    printf("Please enter a character:");
    scanf("%c",&char_x);
    if((char_x >= 'a'&&char_x <= 'm')){
        // n=char_x-a;
        m = char_x + char_x - a + 1;
        // printf("%c",m);
    }
    if(char_x >= 'A'&&char_x <= 'M'){
        // n=char_x-A;
        m = char_x + char_x - A + 1;
        // printf("%c",m);
    }
    if(char_x > 'M'&&char_x <= 'Z'){
        m = char_x + char_x - A - 25;
        // printf("%c",m);
        // 在这里，由于字母表是闭环，所以超过一圈时要减掉一周的量
    }
    if(char_x > 'm'&&char_x <= 'z'){
        m = char_x + char_x - a - 25;
        // printf("%c",m);
    }
    printf("%c",m);
    return 0;
}

```

- 多情况判断类题型

#### 4. 实验四



## 实验四 寻找质数



### ➤ 实验内容

用户输入正整数 $n$ ，找出所有 $2\sim n$ 之间的质数，示例：

对于 $n = 10$ ，则程序输出质数的数量4，并且输出所有质数：2，3，5，7。

### ➤ 注意事项

1. 通过选择和循环语句实现；
2.  $n$ 为正整数且满足 $n \leq 10000$ ；
3. 采用适当的方法提高程序中循环体的执行效率；

### ➤ DDL

1. 3.24号20点之前提交实验报告。
2. 以班级为单位提交至1432302881@qq.com，每人只提交一份pdf（只接受pdf），命名为“姓名+学号”。

```
#include<stdio.h>
#include<math.h>
int main(){
    int n, i, j, k = 1, a[1000];
    printf("please enter a number:\n");
    scanf("%d", &n);
    printf("2 is a prime number.\n");
    for(i = 2; i <= n; i++){
        for(j = 2; j <= i; j++)
        {
            if(i%j == 0)break;
            if(j >= sqrt(i)&& j < sqrt(i) + 1){printf("%d is a prime number.\n", i);
                k++;}
        }
    }
    printf("there are %d prime numbers!", k);
    return 0;
}
```

- 需要注意的是，我们采用的是小于 $n$ 的数依次整除 $n$ 的方法来判断素数，由于 $n = \sqrt{n} * \sqrt{n}$ ，所以我们只需要判断 $1 - \sqrt{n}$ 的数即可，节省计算的时间

## 5. 实验五

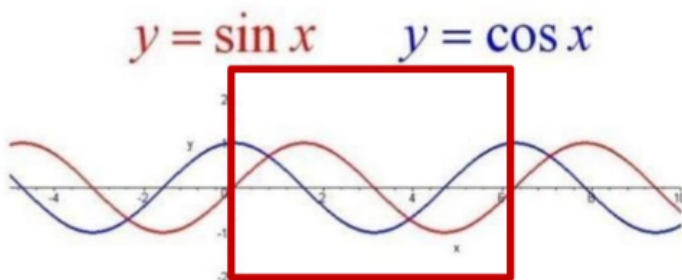


# 实验五 绘制函数曲线



### ➤ 实验内容

绘制 $[0, 2\pi]$ 区间 $\sin(x)$ 和 $\cos(x)$ 的图像，如下图所示：



### ➤ 注意事项

1. 输出的曲线用\*描点构成，绘制形状与上图越接近越好，不需要画出坐标轴；

### ➤ DDL

1. 3.31号20点之前提交

```

#include<stdio.h>
#include<math.h>
int main(){
    double y;
    int xx,xs,xc;
    for(y=1;y>=-1;y-=0.1){
        xc=acos(y)*10;
        xs=asin(y)*10;
        if(xs<0){
            xs=-xs+31;
        }
        for(xx=0;xx<=62;xx++){
            if(xx==xs || xx==31-xs || xx==93-xs){
                printf("*");
            }
            else if(xx==xc || xx==62-xc){
                printf("*");
            }
            else{
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}

```

- 这题考察的是C语言字符串输出的特点，从左到右，从上到下，所以我们只要模拟其过程即可，一行一行扫描，最后绘制成正弦和余弦曲线，在函数上的点为 \* ,不在的为空格

## 6. 实验六



暨南大学  
JINAN UNIVERSITY

## 实验6 线上答题ACM-2

### ➤ 实验内容

### 2. Redraiment的遭遇 21.4%(106/496)1000ms,32M

Submit	My submissions	Statistics		
Description	Input	Output	Input Sample	Output Sample
<p>Redraiment的老家住在工业区，日耗电量非常大。是政府的眼中钉 肉中刺，但又没办法，这里头住的可都是纳税大户呀。今年7月，又传来了不幸的消息，政府要在7、8月对该区进行拉闸限电。但迫于压力，限电制度规则不会太抠门，政府决定从7月1日停电，然后隔一天 到7月3日再停电，再隔两天到7月6日停电，一次下去，每次都比上一次晚一天。Redraiment可是软件专业的学生，怎么离得开计算机。如果停电，就“英雄无用武之地”了。呵呵。所以他开始盘算起自己回家的日子了，他想知道自己到家后到底要经历多少天倒霉的停电。你能帮他算一算吗？</p>				

### ➤ 注意事项

1. 输入包括多组数据,每组数据包括一行:redraiment到家的日期,输入以0/0结束。

例如:

8/31  
0/0

2. 对应每个输入包括一个输出,代表redraiment回家后停电的天数(包括到家那天)。

例如:

0

### ➤ DDL

1. 下课前 (不需要实验报告)



```

#include<stdio.h>
int count(int s);
int main(){
    int m , d, i = 0,n = 1,t = 0, s, output;
    do{
        scanf("%d/%d",&m,&d);
        if(m>8||m<7||n<=0||n>31)return 0;
        s = (m-7)*31+d;
        output = count(s);
        printf("%d\n",output);
    }
    while(m != 0&&d != 0);

}

int count(int s){
    int a[10] = {1,3,6,10,15,21,28,36,45,55};
    int n = 1, t = 0;
    for(int i = 0;i < 10;i++){
        if(a[i] >= s) t++;
    }
    return(t);
}

```

- 这道题需要注意的是，要将日期 date 转化为十进制的数来表示，然后再统计停电的日期

## 7. 实验七



# 实验7 线上答题ACM-95



## ➤ 实验内容

Home Problems A Multiply Problem

### 95. A Multiply Problem 17.6%(6/34)2000ms,32M

Submit My submissions Statistics

Description Input Output Input Sample Output Sample

Our problem is,for a giving M rows and M columns matrix A,output another matrix  $B=A^N$ .

## ➤ 注意事项

input:

```
1
2 2
9 7
5 1
```

output:

```
Case 1:
116 70
50 36
```

## ➤ DDL

1. 下课前（不需要实验报告）

```

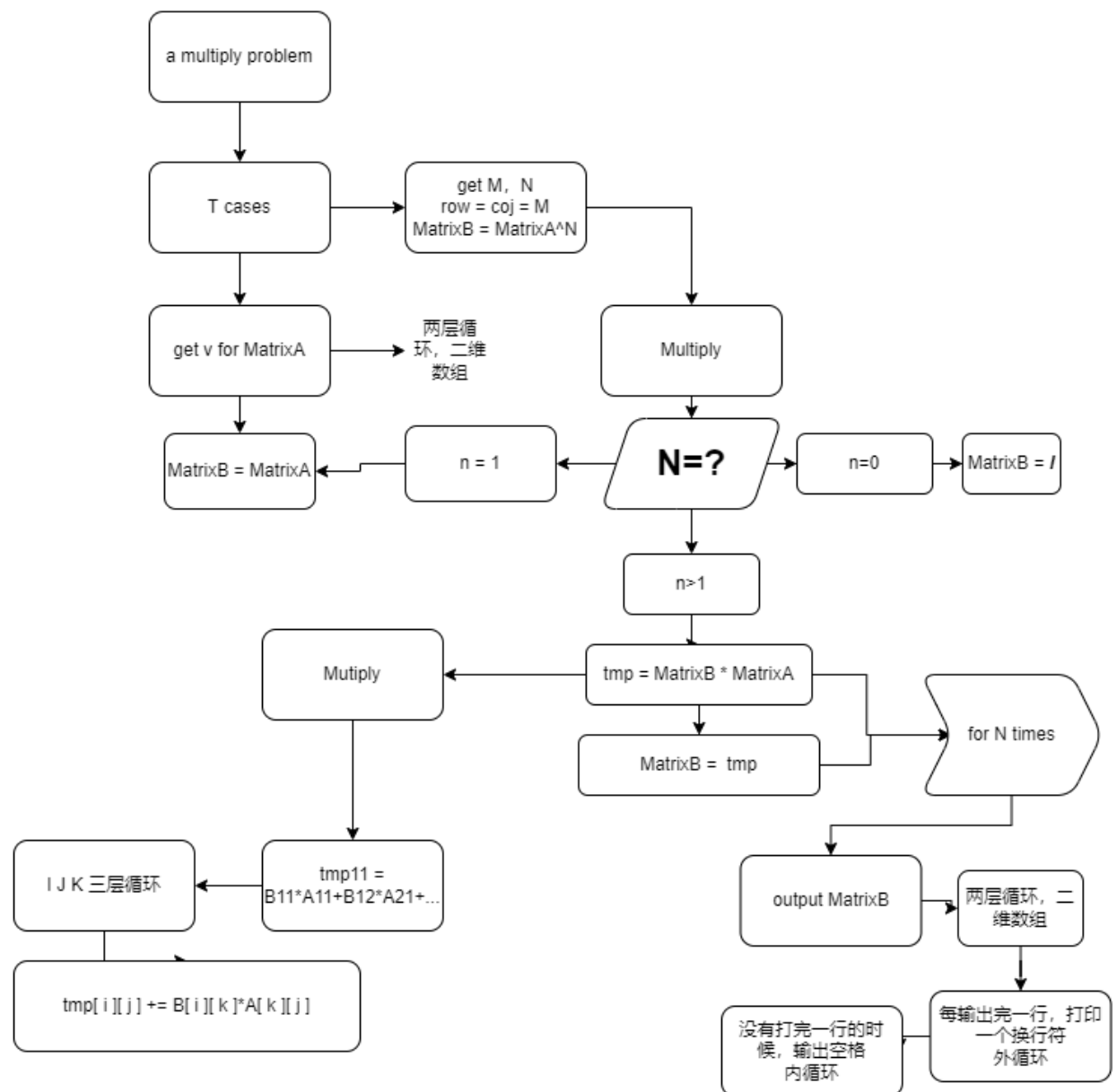
#include<stdio.h>

int main(){
int M,N,T,t,test;
scanf("%d",&T);
for(test=1;test<=T;++test){
    scanf("%d %d",&M,&N);
    int res[4][4]={1,0,0,0},
                {0,1,0,0},
                {0,0,1,0},
                {0,0,0,1}};

    int A[4][4];
    int i,j,t,k;
    for(i=0;i<M;i++){
        for(j=0;j<M;j++){
            scanf("%d",&A[i][j]);
        }
    }
    for(t=0;t<N;++t){
        int tmp[4][4]={0};
        for(i=0;i<M;++i){
            for(j=0;j<M;++j){
                for(k=0;k<M;++k){
                    tmp[i][j]+=res[i][k]*A[k][j];
                }
            }
        }
        for(i=0;i<M;i++){
            for(j=0;j<M;j++){
                res[i][j]=tmp[i][j];
            }
        }
    }
    printf("Case %d:\n",test);
    for(i=0;i<M;i++){
        for(j=0;j<M;j++){
            printf("%d",res[i][j]);
            if(j!=M-1)printf(" ");
        }
        printf("\n");
    }
}
return 0;
}

```

◦ 流程图



- 这题的难点在于手写一个矩阵的乘法运算

## 8. 实验八



## 实验8 线上答题ACM-96



### ➤ 实验内容

#### 96. Be Good at Gaussing 26.7%(4/15)1000ms,32M

[Submit](#) [My submissions](#) [Statistics](#)

[Description](#) [Input](#) [Output](#) [Input Sample](#) [Output Sample](#)

Give you many positive integer  $N$  ( $N \leq 23$ ), for each  $N$ , just output  $N*(N+1)/2$  integers in a single line, separated by space. (Don't ask me why.) For each  $N$ , the output line contains integers from 1 to  $N$ , and each just once. Again, do not ask me why, thank you. I'm so busy. But I can tell you a secret, the output has relationship with number triangle. As: ( $N=3$ )

```
1
2 6
3 4 5
```

See the sample for more information.

### ➤ 注意事项

input:

```
3
4
2
6
```

output:

```
1 2 6 3 4 5
1 2 9 3 10 8 4 5 6 7
1 2 3
1 2 15 3 16 14 4 17 21 13 5 18 19 20 12 6 7 8 9 10 11
```

### ➤ DDL

1. 下课前（不需要实验报告）

```

#include<stdio.h>

int main()
{
    int N,value,row,col,dir,i,j, n;
    while(scanf("%d",&N)!=EOF)
    {
        int tri[24][24]={0};
        row = col = 1;
        value = 1;
        for(;value<=N;++value)
            tri[row++][col]=value;
        row = N;
        col =2 ;
        value = N+1;
        for(;value<=2*N-1;++value)
            tri[row][col++] = value;
        row = N-1;
        col = N-1;
        value = 2*N;
        for(;value<=3*N-3;++value)
            tri[row--][col--] = value;
        row = 2;
        col = 2;
        value = 3*N-2;
        dir = 1;
        while(value<=N*(N+1)/2)
        {
            if(1==dir)
            {
                if((0==tri[row+1][col]))
                    tri[++row][col] = value;
                else
                {
                    dir = 2;
                    tri[row][++col] = value;
                }
            }
            else if(2==dir)
            {
                if((0==tri[row][col+1]))
                    tri[row][++col] = value;
                else
                {
                    dir = 3;
                    tri[--row][--col] = value;
                }
            }
            else
            {

```

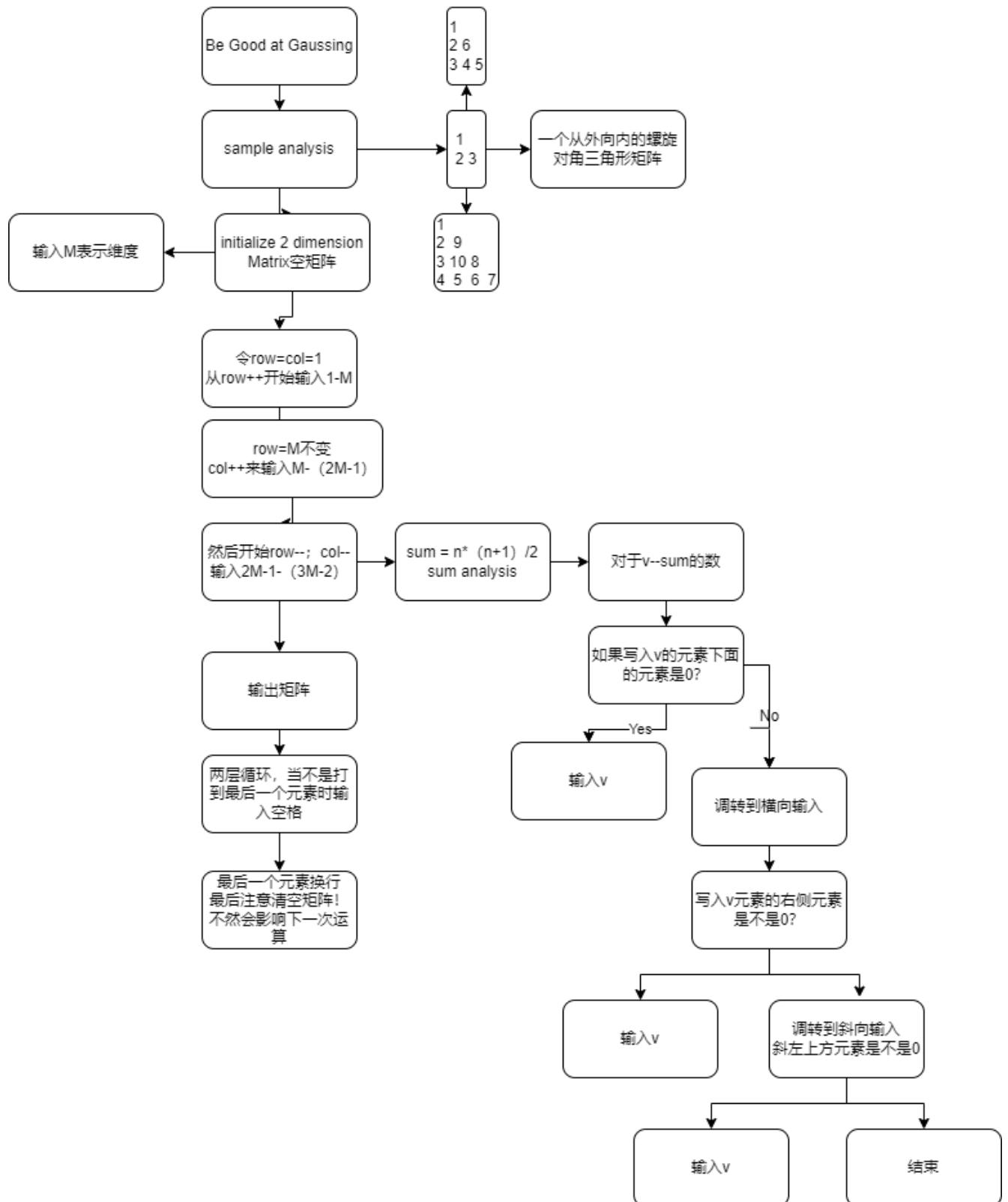
```

        if((0==tri[row-1][col-1]))
            tri[--row][--col] = value;
        else
        {
            dir=1;
            tri[++row][col] = value;
        }
    }
    ++value;
}
row = 1;
for(;row<=N;++row)
{
    col = 1;
    for(;col<=row;++col)
    {
        printf("%d",tri[row][col]);
        if(!(row==N&&col==row))
            {printf(" ");}
        tri[row][col] = 0;
    }
    printf("\n");
}
return 0;
}

```

◦ 流程图

◦



- 模拟过程，这道题的规律是从顶点开始向内RT三角形状螺旋形成二维数组，所以通过一些 选择+判断 即可完成



## 9. 实验九



# 实验9 线上答题ACM-452



### ➤ 实验内容

## 452. 五一放假后遗症 14.3%(3/21)1000ms,32M

[Submit](#) [My submissions](#) [Statistics](#)

[Description](#) [Input](#) [Output](#) [Input Sample](#) [Output Sample](#)

尽管五一放假，雄平同学却是在ACM基地坚持训练。但是他觉得太累了，所以他见到5和1就很生气。贵淳同学想了一个办法，如果能把数里面带1和5的都干掉，那么雄平同学就会感觉舒服多了。但是他们两都很累了，把这个问题推给了陈侠，陈侠毫不犹豫解决了这个问题，但是他也训练了好几天，没空写代码。请帮助ACM三人组写代码计算一个数n内有多少不含1或5的数。

### ➤ 注意事项

input:	5 2 15	output:	3 1 7
--------	--------------	---------	-------------

### ➤ DDL

#### 1. 下课前（不需要实验报告）

```
#include<stdio.h>
int main() {
    int n;
    while (scanf("%d", &n) != EOF) {
        int l = n;
        for (int m = 1; m <= l; m++) {
            int i = m;
            while (i > 0) {
                int j = i % 10;
                i = i / 10;
                if (j == 1 || j == 5) {
                    n = n - 1;
                    break;
                }
            }
        }
        printf("%d\n", n);
    }
}
```

- 思路很简单，循环+从个位数开始判断，遇到 1 or 5 直接 break ,时间复杂度为  $O(n \cdot \log(n))$



# 实验9 线上答题ACM-458



➤ 实验内容

## 458. 计算多项式的值 50%(8/16)1000ms,32M

Submit

My submissions

Statistics

Description

Input

Output

Input Sample

Output Sample

已知 多项式 $f(x)=a_n x^n+....+a_1 x+a_0$ ，求给定 $x_0$ 时函数的值 $f(x_0)$

➤ 注意事项

input:

```
1
2 0
2
1 0
2
3.0 0.0 1.0
1
1.0
```

output:

```
2 0
4
```

➤ DDL

- 1. 下课前（不需要实验报告）

```

#include<stdio.h>
#include<math.h>

int main(){
double f2(int n , double a[] , double x);
int n;
while(scanf("%d",&n)!=EOF){
    double num[10000];
    for(int i = 0;i <= n;++i){
        scanf("%lf",num+i);
    }
    int m;
    scanf("%d",&m);
    double ans[1000];
    for(int j = 0;j < m;++j){
        double x;
        scanf("%lf",x);
        ans[j] = f2(n+1,num,x);
    }
    for(int j = 0;j < m;++j){
        printf("%lf",ans[j]);
        if(j!=m-1)printf(" ");
    }
    printf("\n");
}
}

double f2(int n , double a[] , double x){
    int i;
    double p = a[0];
    for(i = 1;i < n ;i++){
        p += (a[i] * pow(x,i));
    }
    return p;
}

```

- 用**秦九韶算法**更加快速和省内存

## 11. 实验十一



# 实验11 线上答题ACM-465



## ➤ 实验内容

### 465. Olympic Game 14.3%(9/63)1000ms,64M

[Submit](#) [My submissions](#) [Statistics](#)

[Description](#) [Input](#) [Output](#) [Input Sample](#) [Output Sample](#)

2012伦敦奥运会即将到来，大家都非常关注奖牌榜的情况，现在我们假设奖牌榜的排名规则如下：

- 1、首先gold medal数量多的排在前面；
- 2、其次silver medal数量多的排在前面；
- 3、然后bronze medal数量多的排在前面；
- 4、若以上三个条件仍无法区分名次，则以国家名称的字典序排定。

我们假设国家名称不超过20个字符、各种奖牌数不超过100，且大于等于0。

## ➤ 注意事项

input:

```
5
China 32 28 34
England 12 34 22
France 23 33 2
Japan 12 34 25
Rusia 23 43 0
```

output:

```
China
Rusia
France
Japan
England
```

## ➤ DDL

1. 下课前（不需要实验报告）

```

#include<stdio.h>
#include<string.h>
#include<malloc.h>
struct medol
{
    char name[21];
    int jin;
    int yin;
    int tong;
    struct medol *next;
};
struct medol *myhead = NULL;
void printdata();
int my_sort(char *name, int ji, int yi, int to, int len){
int i=0;
struct medol *p=NULL, *q=NULL, *tmp=NULL;
if(myhead == NULL)
    return -1;
if(len == 0)
{
    myhead->next = (struct medol*)malloc(sizeof(struct medol));
    if(myhead->next != NULL)
    {
        memcpy(myhead->next->name, name, 21);
        myhead->next->jin = ji;
        myhead->next->yin = yi;
        myhead->next->tong = to;
        myhead->next->next = NULL;
    }
    else
    {
        return -1;
    }
}
else
{
    if(myhead->next == NULL)
        return -1;

    p = myhead->next;
    tmp = (struct medol*)malloc(sizeof(struct medol));
    if(tmp != NULL)
    {
        memcpy(tmp->name, name, 21);
        tmp->jin = ji;
        tmp->yin = yi;
        tmp->tong = to;
        tmp->next = NULL;
    }
}
}

```

```

else
    return -1;

for(i=0; i< len+1; i++)
{
    if(p != NULL)
    {
        if(ji > p->jin
            || (ji == p->jin && yi > p->yin)
            || (ji == p->jin && yi == p->yin && to > p->tong)
            || (ji == p->jin && yi == p->yin && to == p->tong && strcmp(name, p->name) < 0)
        )
        {
            if(p == myhead->next)
            {
                tmp->next = p;
                myhead->next = tmp;

            }
            else
            {
                tmp->next = p;
                q->next = tmp;
            }
            break;
        }
        else
        {
            q = p;
            p = p->next;
        }
    }
    else
    {
        q->next = tmp;
        break;
    }
}
}
return 0;
}

void printdata()
{
    struct medol *p=NULL;
    if(myhead == NULL)
        return;
    p = myhead->next;
    while(p)

```

```

{
    printf("%s\n", p->name);
    p = p->next;
}
}
void freedata()
{
    struct medol *p=NULL, *q=NULL;
    if(myhead == NULL)
        return;
    p=q=myhead->next;
    while(p)
    {
        q = p->next;
        free(p);
        p = NULL;
        p = q;
    }

    free(myhead);
    myhead = NULL;
}
int main()
{
    int country_num = 0;
    char c_name[21]={0};
    int tmj=0, tmy=0, tmt=0;
    int i=0;

    scanf("%d", &country_num);
    if(country_num > 0 && country_num < 21)
    {

        myhead = (struct medol*)malloc(sizeof(struct medol));
        if(myhead == NULL)
            return 1;
        myhead->next = NULL;

        for(i=0; i<country_num; i++)
        {
            scanf("%s %d %d %d", c_name, &tmj, &tmy, &tmt);
            my_sort(c_name, tmj, tmy, tmt, i);
        }
        printdata();
        freedata();
    }
    return 0;
}

```

- 这道题ZXY提供了两个解题的实现思路：

- 不需要三个变量存储金银铜奖牌数量，用100，10，1的权重计算出权重值再录入即可
- 全部国家输入后再排序会浪费空间而且慢，其实可以输入几个排序几个（**插入排序**）的思想

## 12. 实验十二



# 实验12 双向链表



### ➤ 实验内容

## 465. Olympic Game 14.3%(9/63)1000ms,64M

Submit
My submissions
Statistics

Description
Input
Output
Input Sample
Output Sample

2012伦敦奥运会即将到来，大家都非常关注奖牌榜的情况，现在我们假设奖牌榜的排名规则如下：

- 1、首先gold medal数量多的排在前面；
- 2、其次silver medal数量多的排在前面；
- 3、然后bronze medal数量多的排在前面；
- 4、若以上三个条件仍无法区分名次，则以国家名称的字典序排定。

我们假设国家名称不超过20个字符、各种奖牌数不超过100，且大于等于0。

### ➤ 注意事项

input:

```
5
China 32 28 34
England 12 34 22
France 23 33 2
Japan 12 34 25
Rusia 23 43 0
```

output:

```
China
Rusia
France
Japan
England
```

### ➤ DDL

1. 下课前（不需要实验报告）

- 目前没有好的思路