Fortran 简单程序集

杨洋[1]

!大小写转换
```fortran
program ex39
    implicit none
    character*20 str
    integer i
    print*,'input the string:'
    read*,str
    do i=1,len_trim(str)
        if(str(i:i)>='a'.and.str(i:i)<='z') str(i:i)=char(ichar(str(i:i))-32)
    end do
    print*,'the inverted string:'
    print*,str
end
```

!将字符串转化为整数
```fortran
program ex104
    implicit none
    external f
    character*10 str
    integer f
    print*,'输入由数字组成的字符串:'
    read*,str
    print*,'转化后的整数:'
    print*,f(str)
end
```

```fortran
function f(str)
    implicit none
    character*(*) str
    integer f,k,i
    f=0
    k=len_trim(str)
    do i=1,k
        f=f+(ichar(str(i:i))-ichar('0'))*10**(k-i)
    end do
end function
```

!将十进制数转化为二进制数（用字符串保存）
```fortran
program ex110
```

```fortran
program ex110
    implicit none
    integer a
    character*8::b=' '
    print*,'输入一个十进制整数:'
    read*,a
    do while(a>0)
        if (mod(a,2)==1) then
            b='1'//b
        else
            b='0'//b
        end if
        a=a/2
    end do
    print*,'对应的二进制数为:'
    print*,b
end
```

```fortran
!将二进制数（用字符串保存）转化为十进制数
program ex111
    implicit none
    character*8 a
    integer::b=0,k,i
    print*,'输入一个二进制数:'
    read*,a
    k=len_trim(a)
    do i=1,k
        b=b+(ichar(a(i:i))-ichar('0'))*2**(k-i)
    end do
    print*,'对应的十进制数为:'
    print*,b
end
```

```fortran
!统计大写、小写、数字及其他字符的个数
program ex51
    implicit none
    character*20 str
    integer::i,n1=0,n2=0,n3=0,n4=0
    print*,'输入字符串:'
    read*,str
    do i=1,len_trim(str)
        select case(str(i:i))
            case('a':'z')
                n1=n1+1
```

```fortran
            case('A':'Z')
                n2=n2+1
            case('0':'9')
                n3=n3+1
            case default
                n4=n4+1
        end select
    end do
    print*,'大写字母个数:',n1
    print*,'小写字母个数:',n2
    print*,'数字个数:',n3
    print*,'其他字符个数:',n4
End
```

```fortran
!判断闰年
program ex24
    implicit none
    integer year
    print*,'input a year:'
    read*,year
    if (mod(year,4)==0.and.mod(year,100)/=0.or.mod(year,400)==0) then
        print*,'this is a leap year'
    else
        print*,'this is not a leap year'
    end if
end
```

!求 $e=1+1+\dfrac{1}{2!}+...+\dfrac{1}{n!}$

```fortran
program ex05
    implicit none
    integer::i=1
    real::t=1,sum=0
    do while(abs(t)>=1e-6)
        sum=sum+t
        t=t/i
        i=i+1
    end do
    print*,'e=',sum
end
```
或
```fortran
program ex06
    implicit none
    integer::i=1
```

```fortran
    real::t=1,sum=0
    do
        sum=sum+t
        t=t/i
        i=i+1
        if(abs(t)<1e-6) exit
    end do
    print*,'e=',sum
end
```

!求阶乘n!
```fortran
program ex98
    implicit none
    integer n,i
    integer(8) fact
    print*,'input n:'
    read*,n
    if (n<0) then
        print*,'error!'
    else
        fact=1
        do i=1,n
            fact=fact*i
        end do
        print 100,n,fact
    end if
100 format(1X,I3,'!=',I10)
End
```

!递归法求阶乘
```fortran
program ex70
    implicit none
    external fact
    integer n
    integer(8) fact
    print*,'input n:'
    read*,n
    if (n<0) then
        print*,'error!'
    else
        print 100,n,fact(n)
    end if
    100 format(1X,I3,'!=',I10)
end
```

```
recursive function fact(n)
    implicit none
    integer n
    integer(8) fact
    if (n==0.or.n==1) then
        fact=1
    else
        fact=fact(n-1)*n
    end if
end function
```

!求$\sum_{k=1}^{n} k!$ 从键盘输入 n

```
program ex42
    implicit none
    integer(8)::f=1,sum=0
    integer i,n
    print*,'input n:'
    read*,n
    do i=1,n
        f=f*i
        sum=sum+f
    end do
    print*,'the result:',sum
End
```

!求两个阶乘之和（函数嵌套）
```
program ex69
    implicit none
    external sum
    integer n1,n2
    integer(8) sum
    print*,'input n1,n2:'
    read*,n1,n2
    print*,'the result:',sum(n1,n2)
end
function sum(a,b)
    implicit none
    external fact
    integer a,b
    integer(8) sum,fact
    sum=fact(a)+fact(b)
end function
function fact(n)
```

```fortran
    implicit none
    integer i
    integer(8) fact
    fact=1
    do i=1,n
        fact=fact*i
    end do
end function
```

!求圆周率π，利用公式： $\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} + \cdots + (-1)^{n-1}\frac{1}{(2n-1)}$

```fortran
program ex57
    implicit none
    integer::i=1,sign=1
    real::t=1,pi=0
    do while(abs(t)>=1e-6)
        pi=pi+t
        sign=(-1)*sign
        t=1.0*sign/(2*i+1)
        i=i+1
    end do
    pi=pi*4
    print*,'pi=',pi
end
```

!判断素数
```fortran
program ex07
    implicit none
    integer m,i,k
    print*,'input a number:'
    read*,m
    k=sqrt(real(m))
    do i=2,k
        if(mod(m,i)==0) exit
    end do
    if(i>k) then
        print*,'this is a prime number!'
    else
        print*,'this is not a prime number!'
    end if
End
```
或：
```fortran
program ex99
    implicit none
```

```fortran
    integer n,i,k
    logical::prime=.true.
    print*,'input a number:'
    read*,n
    k=sqrt(real(n))
    i=2
    do while(i<=k)
        if (mod(n,i)==0) then
            prime=.false.
            exit
        end if
        i=i+1
    end do
    if (prime) then
        print*,'this is a prime number!'
    else
        print*,'this is not a prime number!'
    end if
end
```

```fortran
!输出 101 到 200 之间的所有素数
program ex08
    implicit none
    integer::m,k,i,n=0
    do m=101,200,2
        k=sqrt(real(m))
        do i=2,k
            if(mod(m,i)==0) exit
        end do
        if(i>k) then
            print 100,m
            n=n+1
            if(mod(n,5)==0) print*    !每输出 5 个数换行
        end if
    end do
100 format(1X,I5\)
End
```

```fortran
!输出 50-100 的素数并求和（用子程序实现）
program ex71
    implicit none
    external prime
    integer::m,sum=0,n=0
    logical prime
```

```fortran
    do m=51,100,2
        if (prime(m)) then
            sum=sum+m
            print 100,m
            n=n+1
            if (mod(n,5)==0) print*    !每输出 5 个数换行
        end if
    end do
    print*,'sum=',sum
100 format(1X,I3\)
end
function prime(n)
    implicit none
    integer i,k,n
    logical prime
    k=sqrt(real(n))
    do i=2,k
        if (mod(n,i)==0) exit
    end do
    if (i>k) then
        prime=.true.
    else
        prime=.false.
    end if
end function

!找出数组 num 中所有素数，将它们按大小排列在数组 work 中
program ex109
    implicit none
    external prime,sort
    integer,parameter::n=17
    integer::num(n),work(n),k=0,i
    logical prime
    data num/6,19,17,91,2,15,51,11,13,3,25,6,21,14,67,73,72/
    print*,'the original array:'
    print*,(num(i),i=1,n)
    do i=1,n
        if (prime(num(i))) then
            k=k+1
            work(k)=num(i)
        end if
    end do
    call sort(work,k)
    print*,'the final array:'
```

```fortran
        print*,(work(i),i=1,k)
end

function prime(n)
    implicit none
    integer i,k,n
    logical prime
    k=sqrt(real(n))
    do i=2,k
        if (mod(n,i)==0) exit
    end do
    if (i>k) then
        prime=.true.
    else
        prime=.false.
    end if
end function

subroutine sort(a,n)
    implicit none
    integer i,j,k,n,a(n),t
    do i=1,n-1
        k=i
        do j=i+1,n
            if (a(j)>a(k)) k=j
        end do
        if (k/=i) then
            t=a(k)
            a(k)=a(i)
            a(i)=t
        end if
    end do
end subroutine
```

! 求 2～999 中同时满足下列条件的数：
      (a) 该数各位数字之和为奇数；
      (b) 该数是素数。

```fortran
program ex13
    implicit none
    integer::m,m1,m2,m3,k,i,n=0
    do m=2,999
        k=sqrt(real(m))
        do i=2,k
            if(mod(m,i)==0) exit
```

```fortran
        end do
        m1=mod(m,10)
        m2=mod(m/10,10)
        m3=mod(m/100,10)        !或：m3=m/100
        if (i>k.and.mod((m1+m2+m3),2)/=0) then
            print 100,m
            n=n+1
            if(mod(n,5)==0) print*     !每输出 5 个数换行
        end if
    end do
100 format(1X,I3\)
End
```

!验证哥德巴赫猜想（任何充分大的偶数都可表示为两个素数之和）

```fortran
program ex02
    implicit none
    external prime
    integer m,m1,k1,k2,i,j
    logical prime
    do m=4,20,2
        do m1=2,m-2
            if (prime(m1).and.prime(m-m1)) print 100,m,m1,m-m1
        end do
    end do
100 format(1X,I3,'=',I2,'+',I2)
end
function prime(n)
    implicit none
    integer i,k,n
    logical prime
    k=sqrt(real(n))
    do i=2,k
        if (mod(n,i)==0) exit
    end do
    if (i>k) then
        prime=.true.
    else
        prime=.false.
    end if
end function

program ex112
    implicit none
    external prime
```

```fortran
    integer::n,i,a(4),sum
    logical prime
    do n=1000,9999
        sum=0
        do i=1,4
            a(5-i)=mod(n/10**(i-1),10)
        end do
        do i=1,4
            sum=sum+a(i)
        end do
        if (prime(n).and.mod(sum,2)/=2) print*,n
    end do
end
function prime(n)
    implicit none
    integer i,k,n
    logical prime
    k=sqrt(real(n))
    do i=2,k
        if (mod(n,i)==0) exit
    end do
    if (i>k) then
        prime=.true.
    else
        prime=.false.
    end if
end function
```

!求 3～39 之间满足下列条件的各组素数：每组有三个素数，第二个比第一个大 2，第三个比第二个大 4。输出满足条件的所有解并求出这个样的素数共有多少组。

```fortran
program ex03
    implicit none
    external prime
    integer::m,n=0
    logical prime
    do m=3,39
        if (prime(m).and.prime(m+2).and.prime(m+6)) then
            n=n+1
            print*,m,m+2,m+6
        end if
    end do
    print*,'共有',n,'组这样的素数'
end
function prime(n)
```

```fortran
    implicit none
    integer i,k,n
    logical prime
    k=sqrt(real(n))
    do i=2,k
        if (mod(n,i)==0) exit
    end do
    if (i>k) then
        prime=.true.
    else
        prime=.false.
    end if
end function
```

!求 2～10000 之间所有完数（除自身之外的所有因子之和等于自身的数）
```fortran
program ex14
    implicit none
    integer::m,i,sum,n=0
    do m=2,10000
        sum=0
        do i=1,m-1      !除自身之外
            if (mod(m,i)==0) sum=sum+i
        end do
        if (sum==m) then
            print 100,m
            n=n+1
            if (mod(n,5)==0) print*    !每输出 5 个数换行
        end if
    end do
100 format(1X,I5\)
End
```

!求 2～10000 之间所有的同构数（如：$5^2 = 25, 25^2 = 625$）
```fortran
program ex15
    implicit none
    integer::m,tmp1,tmp2,n=0
    print*,'2-10000 之间的所有同构数:'
    do m=2,10000
        tmp1=m
        tmp2=m**2
        do while(mod(tmp1,10)==mod(tmp2,10).and.tmp1/=0)
            tmp1=tmp1/10
            tmp2=tmp2/10
        end do
```

```
        if (tmp1==0) then
            print 100,m
            n=n+1
            if (mod(n,5)==0) print*
        end if
    end do
    print*
100 format(1X,I4\)
End
```

!求水仙花数（如$153 = 1^3 + 5^3 + 3^3$）
```
program ex36
    implicit none
    integer::m,m1,m2,m3,n=0
    do m=100,999
        m1=mod(m,10)
        m2=mod(m/10,10)
        m3=mod(m/100,10)        !或：m3=m/100
        if ((m1**3+m2**3+m3**3)==m) then
            print 100,m
            n=n+1
            if (mod(n,5)==0) print*
        end if
    end do
100 format(1X,I3\)
End
```

!矩形法求定积分$\int_a^b \sin x \, dx$
```
program ex10
    implicit none
    integer i,n
    real::a,b,h,f,si,s=0
    print*,'输入积分上下限:'
    read*,a,b
    print*,'输入分割数:'
    read*,n
    h=(b-a)/n
    do i=1,n
        f=sin(a+(i-1)*h)
        si=f*h
        s=s+si
    end do
    print*,'定积分结果:',s
```

end

!梯形法求定积分 $\int_a^b \sin x \, dx$

```
program ex31
    implicit none
    real::a,b,h,si,f1,f2,s=0
    integer i,n
    print*,'输入积分上下限:'
    read*,a,b
    print*,'输入分割数:'
    read*,n
    h=(b-a)/n
    do i=1,n
        f1=sin(a+(i-1)*h)
        f2=sin(a+i*h)
        si=(f1+f2)*h/2
        s=s+si
    end do
    print*,'定积分结果:',s
end
```

!求 Fibonacci 数列
```
program ex12
    implicit none
    integer f1,f2,f,i,n
    print*,'input n:'
    read*,n
    f1=1;f2=1
    print 100,f1,f2
    do i=3,n
        f=f1+f2
        print 200,f
        f1=f2
        f2=f
        if(mod(i,5)==0) print*
    end do
100 format(1X,I5,1X,I5\)
200 format(1X,I5\)
End
```
或利用数组实现：
```
program ex54
    implicit none
    integer,parameter::n=40
```

```
    integer f(n),i
    f(1)=1
    f(2)=1
    do i=3,n
        f(i)=f(i-2)+f(i-1)
    end do
    print 100,(f(i),i=1,n)
100 format(1X,5I10)
end
```

!求 Fibonacci 数列大于 4000 的最小项及 5000 之内的项数
```
program ex04
    implicit none
    integer::f,f1,f2,i,n=2
    f1=1
    f2=1
    do while(f2<5000)
        f=f1+f2
        f1=f2
        f2=f
        n=n+1
        if (f1<4000.and.f2>4000) print*,'大于 4000 的最小值:',f1
    end do
    print*,'5000 之内的项数:',n
end
```

!最大公约数和最小公倍数（辗转相除法）
```
program ex16
    implicit none
    integer m,n,r,tmp1,tmp2,t
    print*,'input two integers:'
    read*,m,n
    if (m<n) then
        t=m
        m=n
        n=t
    end if
    tmp1=m
    tmp2=n
    r=mod(tmp1,tmp2)
    do while(r/=0)
        tmp1=tmp2
        tmp2=r
        r=mod(tmp1,tmp2)
```

```
    end do
    print*,'GCD:',tmp2
    print*,'LCM:',m*n/tmp2
end
```

!九九乘法表下三角
```
program ex17
    implicit none
    integer i,j,t
    do i=1,9
        do j=1,i
            t=i*j
            print 100,t
        end do
    print*
    end do
100 format(1X,I5\)
End
```
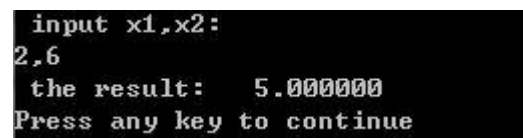
!二分法求根$x^3 - 5x^2 + 16x - 80 = 0$，输入的两个根必须在真实根两侧
```
program ex39
    implicit none
    external f
    real x0,x1,x2,f
    do
        print*,'input x1,x2:'
        read*,x1,x2
        if (f(x1)*f(x2)<0) exit
    end do
    do
        x0=(x1+x2)/2
        if (f(x0)*f(x1)>0) then
            x1=x0
        else
            x2=x0
        end if
        if (abs(f(x0))<1e-6) exit
    end do
    print*,'the result:',x0
end
function f(x)
    implicit none
    real x,f
    f=x**3-5*x**2+16*x-80
```

```
end function
```

!弦截法，求$x^3 - 5x^2 + 16x - 80 = 0$ 的根，其中$X_0 = \dfrac{x1f(x2)-x2f(x1)}{f(x2)-f(x1)}$

```
program ex83
    implicit none
    external f
    real x1,x2,f,x0
    do
        print*,'input x1,x2:'
        read*,x1,x2
        if (f(x1)*f(x2)<0) exit
    end do
    do
        x0=(x1*f(x2)-x2*f(x1))/(f(x2)-f(x1))
        if (f(x0)*f(x1)>0) then
            x1=x0
        else
            x2=x0
        end if
        if (abs(f(x0))<1e-6) exit
    end do
    print*,'the result:',x0
 end
 function f(x)
    implicit none
    real x,f
    f=x**3-5*x**2+16*x-80
end function
```

```
 input x1,x2:
2,6
 the result:    5.000000
Press any key to continue
```

!牛顿迭代法求根$x^3 - 5x^2 + 16x - 80 = 0$
```
program ex20
    implicit none
    external f,g        !g(x)是 f(x)的导函数
    real xr,xtry,f,g
    print*,'input xtry:'
    read*,xtry
    do
        xr=xtry-f(xtry)/g(xtry)
        if(abs(f(xr))<1e-6) exit
        xtry=xr
```

```fortran
        end do
        print*,'result:',xr
end
function f(x)
        implicit none
        real f,x
        f=x**3-5*x**2+16*x-80
end function
function g(x)
        implicit none
        real g,x
        g=3*x**2-10*x+16
end function
```

!求矩阵每行元素平均值、最大值及最小值
```fortran
program ex23
        implicit none
        integer,parameter::m=3,n=4
        integer i,j
        real sum,aver,max,min,a(m,n)
        print*,'输入二维数组:'
        read*,((a(i,j),j=1,n),i=1,m)
        do i=1,m
            sum=0
            max=a(i,1)
            min=a(i,1)
            do j=1,n
                sum=sum+a(i,j)
                if (a(i,j)>max) max=a(i,j)
                if (a(i,j)<min) min=a(i,j)
            end do
            aver=sum/n
            print 100,i,max,min,aver
        end do
100 format(1X,'第',I2,'行',2X,'最大值:',F5.1,2X,'最小值:',F5.2,2X,'平均值:',F5.2)
End
```

!顺序查找
```fortran
program ex26
        implicit none
        integer,parameter::n=10
        integer::a(n),i,index,flag=0
        print*,'input numbers:'
        read*,(a(i),i=1,n)
```

```fortran
        print*,'input index:'
        read*,index
        do i=1,n
            if (a(i)==index) then
                flag=1
                print 100,index,i
            end if
        end do
        if (flag==0) print*,'not found!'
100 format(1X,I5,'is No.',I5)
End
```

!折半法查找
```fortran
program ex27
        implicit none
        integer,parameter::n=10
        integer::a(n),index,top=1,bottom=10,mid,i
        logical::found=.false.
        print*,'enter the numbers:'
        read*,(a(i),i=1,n)
        print*,'input index:'
        read*,index
        do while(top<=bottom.and.found==.false.)
            mid=(top+bottom)/2
            if (index<a(mid)) then
                bottom=mid-1
            else if (index>a(mid)) then
                top=mid+1
            else
                found=.true.
                print 100,index,mid
            end if
        end do
        if (.not.found) print*,'not found!'
100 format(1X,I5,'is located at No.',I5)
End
```

!三个数从小到大排序
```fortran
program ex66
        implicit none
        external swap
        integer a,b,c
        print*,'input three numbers:'
        read*,a,b,c
```

```fortran
    if (a>b) call swap(a,b)
    if (a>c) call swap(a,c)
    if (b>c) call swap(b,c)
    print*,'the result:',a,b,c
end
subroutine swap(x,y)
    implicit none
    integer x,y,t
    t=x
    x=y
    y=t
end subroutine
```

!选择法从大到小排序
```fortran
program ex77
    implicit none
    external sort
    integer,parameter::n=10
    integer i
    real a(n)
    print*,'input numbers:'
    read*,(a(i),i=1,n)
    call sort(a,n)
    print*,'the sorted array:'
    print*,(a(i),i=1,n)
end

subroutine sort(a,n)
    implicit none
    integer i,j,k,n
    real a(n),t
    do i=1,n-1
        k=i
        do j=i+1,n
            if (a(j)>a(k)) k=j
        end do
        if (k/=i) then
            t=a(k)
            a(k)=a(i)
            a(i)=t
        end if
    end do
end subroutine
```
子程序也可写为：

```
subroutine sort(a,n)
    implicit none
    integer i,j,n,
    real a(n),t
    do i=1,n-1
        do j=i+1,n
            if (a(j)>a(i)) then
                t=a(i)
                a(i)=a(j)
                a(j)=t
            end if
        end do
    end do
end subroutine
```

!冒泡法排序
```
program ex44
    implicit none
    external sort
    integer,parameter::n=10
    integer i,
    real a(n)
    print*,'input numbers:'
    read*,(a(i),i=1,n)
    call sort(a,n)
    print*,'the sorted array:'
    print*,(a(i),i=1,n)
end
subroutine sort(a,n)
    implicit none
    integer i,j,n
    real a(n),t
    do j=1,n-1
        do i=1,n-j
            if (a(i)<a(i+1)) then
                t=a(i)
                a(i)=a(i+1)
                a(i+1)=t
            end if
        end do
    end do
end subroutine
```

!逆序

```
program ex82
    implicit none
    external inv
    integer,parameter::n=10
    integer::a(n),i
    print*,'input the array:'
    read*,(a(i),i=1,n)
    call inv(a,n)
    print*,'the inverted array:'
    print*,(a(i),i=1,n)
end
subroutine inv(a,n)
    implicit none
    integer n,i,j,
    integer a(n),t
    do i=1,n/2
        j=n+1-i
        t=a(i)
        a(i)=a(j)
        a(j)=t
    end do
end subroutine
```

!判断回文（如 RaDar 是回文，不区分大小写）
```
program ex105
    implicit none
    character*10 str
    integer i,k,left,right
    print*,'input the string:'
    read*,str
    k=len_trim(str)
    do i=1,k/2
        left=ichar(str(i:i))
        right=ichar(str(k+1-i:k+1-i))
        if (left/=right.and.abs(left-right)/=32) exit
    end do
    if (i>k/2) then
        print*,'this is a palindrom!'
    else
        print*,'this is not a palindrom!'
    end if
end
```

!转置

```
program ex29
    implicit none
    integer,parameter::m=4,n=3
    integer a(m,n),b(n,m),i,j
    print*,'enter the matrix:'
    read*,((a(i,j),j=1,n),i=1,m)
    do i=1,m
        do j=1,n
            b(j,i)=a(i,j)
        end do
    end do
    print 100,((b(i,j),j=1,m),i=1,n)
    100 format(1X,<m>I5)
End
```
不引入新数组只能针对方阵
```
program ex40
    implicit none
    integer,parameter::n=3
    integer a(n,n),i,j,t
    print*,'enter array a:'
    read*,((a(i,j),j=1,n),i=1,n)
    do i=1,n
        do j=1,i        !或 do j=1,i-1
            t=a(i,j)
            a(i,j)=a(j,i)
            a(j,i)=t
        end do
    end do
    print 100,((a(i,j),j=1,n),i=1,n)
    100 format(1X,<n>I5)
End
```

```
!输出杨辉三角
program ex46
    implicit none
    integer,parameter::n=10
    integer a(n,n),i,j
    do i=1,n
        a(i,1)=1
        a(i,i)=1
    end do
    do i=3,n
        do j=2,i-1
            a(i,j)=a(i-1,j-1)+a(i-1,j)
```

```fortran
        end do
    end do
    print*,'the Yang Hui Triangular Matrices:'
    print 100,((a(i,j),j=1,i),i=1,n)
100 format(1X,<i>I5)
End
```

!矩阵乘法
```fortran
program ex30
    implicit none
    integer,parameter::m=3,n=4,p=2
    integer a(m,n),b(n,p),c(m,p)
    integer i,j,k
    print*,'enter array a:'
    read*,((a(i,j),j=1,n),i=1,m)
    print*,'enter array b:'
    read*,((b(i,j),j=1,p),i=1,n)
    do i=1,m
        do j=1,p
            c(i,j)=0
            do k=1,n
                c(i,j)=c(i,j)+a(i,k)*b(k,j)
            end do
        end do
    end do
    print*,'the result:'
    print 100,((c(i,j),j=1,p),i=1,m)
100 format(1X,<p>I5)
End
```

!插入（数组是从大到小排好序的）
```fortran
program ex32
    implicit none
    integer a(100),i,n,insert_num,p
    print*,'input n:'
    read*,n
    print*,'input numbers:'
    read*,(a(i),i=1,n)
    print*,'enter insert_num:'
    read*,insert_num
    p=1
    do while (insert_num>a(p).and.p<=n)
        p=p+1
    end do
```

```
    do i=n,p,-1
        a(i+1)=a(i)
    end do
    a(p)=insert_num
    print*,'the inserted array:'
    print*,(a(i),i=1,n+1)
end
```

!删除（把与输入相同的数都删去）
```
program ex73
    implicit none
    integer i,n,delete_num,p
    integer,allocatable::a(:)
    print*,'input n:'
    read*,n
    allocate(a(n))
    print*,'input numbers:'
    read*,(a(i),i=1,n)
    print*,'input the delete_num:'
    read*,delete_num
    p=1
    do
        do while(delete_num/=a(p).and.p<=n)
            p=p+1
        end do
        if (a(p)==delete_num) then
            do i=p,n-1
                a(i)=a(i+1)
            end do
            n=n-1
        end if
        if (p>n) exit
    end do
    print*,'the deleted array:'
    print*,(a(i),i=1,n)
    deallocate(a)
end
```
或：
```
rogram ex107
    implicit none
    integer::i,n,delete_num,p,count=0
    integer,allocatable::a(:)
    print*,'input n:'
    read*,n
```

```fortran
    allocate(a(n))
    print*,'input numbers:'
    read*,(a(i),i=1,n)
    print*,'input the delete_num:'
    read*,delete_num
    p=1
    do
        do while(delete_num/=a(p).and.p<=n-count+1)
            p=p+1
        end do
        if (a(p)==delete_num) then
        count=count+1
            do i=p,n-count
                a(i)=a(i+1)
            end do
        end if
        if (p>n-count+1) exit
    end do
    print*,'the deleted array:'
    print*,(a(i),i=1,n-count)
    deallocate(a)
end
```

!去空格
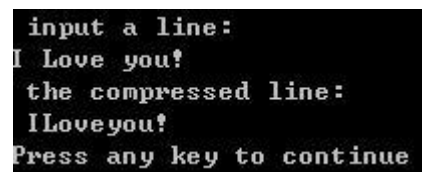```fortran
program ex106
    implicit none
    character*50 line
    integer::i,j,k,count=0
    print*,'input a line:'
    read 100,line   !必须要用有格式输入
    i=1
    k=len_trim(line)
    do
        if (line(i:i)/=' ') then
            i=i+1
            cycle
        else if (line(i:i)==' ') then
            count=count+1
            do j=i,k-count
                line(j:j)=line(j+1:j+1)
            end do
            if (i>=k+1-count) exit
            line(k+1-count:k+1-count)=' '
        end if
```

```
        end do
        print*,'the compressed line:'
        print*,trim(line)
100 format(A)
End
```
或：
```
program ex106
    implicit none
    character*50 line
    integer::i,j,k,count=0
    print*,'input a line:'
    read 100,line
    i=1
    k=len_trim(line)
    do
        do while(line(i:i)/=' '.and.i<=k+1-count)
            i=i+1
        end do
        if (line(i:i)==' ') then
            count=count+1
            do j=i,k-count
                line(j:j)=line(j+1:j+1)
            end do
        end if
        if (i>k+1-count) exit
        line(k+1-count:k+1-count)=' '
    end do
    print*,'the compressed line:'
    print*,trim(line)
100 format(A)
end
```



```
!删除相同的数到只剩一个(只留第一个)
program ex72
    implicit none
    integer i,j,p,n
    integer,allocatable::a(:)
    print*,'input n:'
    read*,n
    allocate(a(n))
```

```fortran
    print*,'input numbers:'
    read*,(a(i),i=1,n)
    i=1    !不能用 do 循环，因为 do 循环的循环次数在一开始就已经算好了
    do while(i<=n)
        p=i+1
        do
            do while(a(i)/=a(p).and.p<=n)
                p=p+1
            end do
            if (a(i)==a(p)) then
                do j=p,n-1
                    a(j)=a(j+1)
                end do
                n=n-1
            end if
            if (p>n) exit
        end do
        i=i+1
    end do
    print*,'the final array:'
    print*,(a(i),i=1,n)
    deallocate(a)
end
```

或：

```fortran
program ex84
    implicit none
    integer,parameter::n=10
    integer::a(n),b(n),i,j,k=1
    print*,'input numbers:'
    read*,(a(i),i=1,n)
    b(1)=a(1)
    do i=2,n
        do j=1,k
            if (a(i)==b(j)) exit
        end do
        if (a(i)/=b(j)) then
            b(k+1)=a(i)
            k=k+1
        end if
    end do
    print*,'the final array:'
    print*,(b(i),i=1,k)
end
```

!读入若干字符行，输出其中最长的一行字符串（用字符数组处理）

```fortran
program ex79
    implicit none
    integer i,n,k,max
    character*20,allocatable::str(:)
    print*,'input n:'
    read*,n
    allocate(str(n))
    print*,'input the strings:'
    read*,(str(i),i=1,n)
    max=len_trim(str(1))
    do i=1,n
        if (len_trim(str(i))>max) then
            max=len_trim(str(i))
            k=i
        end if
    end do
    print*,'the longest row:',str(k)
    deallocate(str)
end
```

!统计每位候选人的选票数及所占比例

```fortran
program ex94
    implicit none
    type person
        character*20,name
        integer count
    end type person
    integer,parameter::n=10
    integer i,j,k

type(person)::leader(5)=(/person('Li',0),person('Yang',0),person('Zhang',0),person('Ren',0),person('Wu',0)/),t
    character*20 leader_name
    do i=1,n
        read*,leader_name
        do j=1,5
            if (leader_name==leader(j).name) leader(j).count=leader(j).count+1
        end do
    end do
    do i=1,4
        k=i
        do j=i+1,5
            if (leader(j).count>leader(k).count) k=j
```

```fortran
        end do
        if (k/=i) then
            t=leader(i)
            leader(i)=leader(k)
            leader(k)=t
        end if
    end do
    print*,'the result:'
    print 100,(leader(i).name,leader(i).count,1.0*leader(i).count/n*100,i=1,5)
100 format(1X,A5,':',I3,',',F4.1,'%')
End
```

学号	数学	英语	物理	Fortran
A1001	98	96	90	91
A1002	56	50	58	45
A1003	90	72	82	92

用 data 语句将上表中的数据分别赋值给数组 num 和 score，要求

1、 输出成绩并保存到 source.txt 文件中

2、 输出单科成绩最高学生的学号和成绩并保存到 max.txt 文件中

3、 计算每位学生的平均成绩，输出平均分不及格学生的学号和 4 门课成绩并统计不及格人数

4、 按平均分高低排序，输出结果并保存到 aver.txt 文件中

5、 从 source.txt 文件中读取所有学生的数据并输出学生学号和 Fortran 成绩，保存到 fortran.txt 文件中

```fortran
program ex108
    implicit none
    external sort
    integer,parameter::n=3
    character*5 num(n)
    integer::i,j,score(n,4),max,row=1,sum,count=0,aver(n)
    data num/'A1001','A1002','A1003'/
    data score/98,56,90,96,50,72,90,58,82,91,45,92/          !按列存储

    print*,'该班的成绩表:'
    print*,'学号  数学  英语  物理  Fortran'
    print '(1X,A5,4I5)',(num(i),(score(i,j),j=1,4),i=1,n)
    open(10,file='souce.txt')
    write(10,'(1X,A5,4I5)') (num(i),(score(i,j),j=1,4),i=1,n)
    print*

    print*,'各科最高分学生的学号及成绩:'
    print*,'学号      成绩:'
    open(20,file='max.txt')
```

```fortran
do j=1,4
    max=score(1,j)
    do i=2,n
        if (score(i,j)>max) then
            max=score(i,j)
            row=i
        end if
    end do
    print '(1X,A5,I5)',num(row),max
    write(20,'(1X,A5,I5)') num(row),max
end do
close(20)
print*

print*,'平均分不及格学生的学号及四门课成绩:'
print*,'学号 数学 英语 物理 Fortran 平均分'
do i=1,n
    sum=0
    do j=1,4
        sum=sum+score(i,j)
    end do
    aver(i)=sum/4
    if (aver(i)<60) then
        count=count+1
        print '(1X,A5,5I5)',num(i),(score(i,j),j=1,4),aver(i)
    end if
end do
print*,'共有',count,'个学生不及格'
print*

call sort(aver,num,n)
print*,'按平均分从高到低排序后的学生数据:'
print '(1X,A5,I5)',(num(i),aver(i),i=1,n)
open(30,file='aver.txt')
write(30,'(1X,A5,I5)') (num(i),aver(i),i=1,n)
close(30)
print*

rewind(10)
open(40,file='fortran.txt')
print*,'输出所有学生的 Fortran 成绩:'
read(10,'(1X,A5,4I5)') (num(i),(score(i,j),j=1,4),i=1,n)
print*,'学号      Fortran'
print '(1X,A5,I5)',(num(i),score(i,4),i=1,n)
```

```fortran
        write(40,'(1X,A5,I5)') (num(i),score(i,4),i=1,n)
        close(40)
        close(10)
end

subroutine sort(a,num,n)
    implicit none
    integer i,j,k,n
    integer a(n),t
    character*5 num(n),p
    do i=1,n-1
        k=i
        do j=i+1,n
            if (a(j)>a(k)) k=j
        end do
        if (i/=k) then
            t=a(i)
            a(i)=a(k)
            a(k)=t
            p=num(i)
            num(i)=num(k)
            num(k)=p
        end if
    end do
end subroutine
```