

Fortrani programmid.

Programm 1

```
print *, "hello world"
stop
end
```

Programm 2

```
real a,b,c      loome reaalmuutujad a,b ja c
read *,a,b      loome a ja b klaviatuurist
c=a+b          võrdlustehted aritmeetikatehted
print *, "a+b=",c
stop
end
```

Programm 3

```
real a,b,c      loome reaalmuutujad a,b ja c
open(10,file="test3.in")  avame faili numbriga 10 ja nimega test3.in
open(11,file="test3.out") avame faili numbriga 11 ja nimega test3.out
read(10,*) a,b    loome a ja b esimest failist
c=a+b            summerime a ja b
print *, "a+b=",c  tulemuse kirjutamine ekraanile
write(11,*) c      salvestame tulemuse faili
stop
end
```

Programm 4 massivi elementide liitmine

```
real v(7),sum    loodud reaalmuutuja sum ja reaalmassiv v pikkusega 7
integer n        loodud täismuutuja n
open(10,file="test4.in")
read(10,*) n     loeme massiivi elementide arv. Ei pea ületada 7 (fikseeritud massiv)
read(10,*) v     loeme failist massivi elementid
! read(10,*) (v(i),i=1,n) ekvivalentne vorm
sum=0.           sum muutuja initsialiseerimine
do i=1,n         tsükli operaator
    sum=sum+v(i) elementide liitmine
enddo            tsükli lõpp
print*, "sum=", sum, "keskm.=", sum/n
stop
end
```

Programm 4.1 massivi elementide liitmine

```
real,allocatable:: v(:),sum  loome tegelik muutuja "sum" ja ühemõõtmelist dünaamilist
!                             massiivi v
integer n                    loodud täismuutuja n
open(10,file="test4.in")
read(10,*) n                 loeme massiivi elementide arv.
allocate(v(n))               loome tegelik massivi
read(10,*) v                 loeme failist massivi elementid
! read(10,*) (v(i),i=1,n) ekvivalentne vorm
sum=0.                       sum muutuja initsialiseerimine
do i=1,n                     tsükli operaator
    sum=sum+v(i)             elementide liitmine
```

```

enddo                                tsükli lõpp
print*, "sum=", sum, "keskm.=", sum/n
stop
end

```

Programm 5 otsing maksimaalset ja minimaalset arvud

```

real a(10)
integer n
open(10,file="test5.in")
read(10,*) n
read(10,*) (A(i),i=1,n)
amax = -1.E+32
amin = 1.E+32
do i = 1,n
if(a(i).gt.amax)then
    amax=a(i)                jälgima,vaatlema
endif
if(a(i).lt.amin)then
    amin=a(i)
endif
enddo
print*, "max=", amax, "min=", amin
stop
end

```

Programm 6 faktoriaali arvutamine

```

integer n,i,fakt
read *,n
if(n.lt.0)then    valikulause, valiju tingimuse tõde vale
    print *, "n on neg."
stop
endif
if(n.eq.0)then
    fakt=1
    print *, "", fakt
stop
endif
if(n.lt.0)then
    print *, "n on neg."
return
endif
fakt=1
do i=1,n
    fakt=fakt*i
enddo

print *, "fakt=", fakt
stop
end

```

Programm 7 *maatriksite korrutamine*

```
real,allocatable:: a(:,,:),b(:,,:),c(:,,:)
open(10,file="test7.in")
open(20,file="test7.out")
read(10,*) n
allocate(a(n,n),b(n,n),c(n,n))
read (10,*) a,b
do i=1,n
do j=1,n
    sum=0.
do k=1,n
    sum=sum+a(i,k)*b(k,j)
enddo
    c(i,j)=sum
enddo
enddo

write (20,100) C
100 format(4F10.4)
stop
end
```

Programm 8 *leidmine ruutvõrrandi juured*

```
real a,b,c
read *,a,b,c
diskr= b*b-4.*a*c

if(diskr.lt.0)then
    x1=(-b+sqrt(-diskr))/(2.*a)
    x2=(-b-sqrt(-diskr))/(2.*a)
    print *, "root on complex = ",x1,"+i",x2,x1,"-i",x2
stop
endif

if(diskr.eq.0)then
    x1=-b/(2.*a)
    print*, "root on=",x1
stop
endif

if(diskr.gt.0)then
    x1=(-b+sqrt(diskr))/(2.*a)
    x2=(-b-sqrt(diskr))/(2.*a)
    print*, "root on=",x1," ",x2
stop
endif

end
```

Programm 9 **kesk** *alamprogrammi kasutamine massiivi elementide liitmiseks*

```

real v(10),sum
integer n
open(11,file="test9.in")
read(11,*) n
read(11,*) v
call kesk(n,v)
print*, "sum=",sum,"keskm.=", sum/n
stop
end

```

```

subroutine kesk(n,a)
real a(n)
sum=0.
do i=1,n
    sum=sum+a(i)
enddo
return
end

```

Programm 10 **mult** *alamprogrammi kasutamine matriksite korrutamiseks*

```

real,allocatable:: a(:,,:),b(:,,:),c(:,,:)
open(10,file="test10.in")
open(20,file="test10.out")
allocate(a(n,n),b(n,n),c(n,n))
read (10,*) a,b
call mult(n,a,b,c)
write (20,100) c
100 format(4F10.4)
stop
end

```

```

subroutine mult(n,a,b,c)
real a(n,n),b(n,n),c(n,n)
do i=1,n
do j=1,n
    sum=0.
do k=1,n
    sum=sum+a(i,k)*b(k,j)
enddo
    c(i,j)=sum
enddo
enddo
return
end

```

Programm 11 **fact** *alamprogrammi kasutamine faktoriaali korrutamiseks*

```

integer n
read *,n

```

```

    call fact(n,ifact)
    print 100,ifact
100  format("fakt=",I10)
    end

    subroutine fact(n,ifact)

    if(n.lt.0)then
        print *,"n on neg."
        stop
    endif

    if(n.eq.0)then
        ifact=1
    return
    endif

    ifact=1
    do i=2,n
        ifact=ifact*i
    enddo

    return
    end

```

Programm 12 **maxmin** *alamprogrammi kasutamine maksimaalsete ja minimaalsete arvude leidmiseks*

```

    real a(10)
    integer n
    open(10,file="test7.in")
    Read(10,*) n
    Read(10,*) (a(i),i=1,n)
    call maxmin(n,a,amax,amin)
    print*,"max=",amax,"min=",amin
    stop
    end
    subroutine maxmin(n,a,amax,amin)
    real a(n)
    amax = -1.E+32
    amin = 1.E+32
    do i =1,n
    if(a(i).gt.amax)then
        amax=a(i)
    endif
    if(a(i).lt.amin)then
        amin=a(i)
    endif
    enddo

```

```
return  
end
```

Programm 13 $\int_0^1 x^2 dx$ integraali arvutamine lihtmeetodil

```
b=1.  
a=0.  
n=1000  
dx=(b-a)/n  
sum=0.  
do i=1,n-1  
  x=a+i*dx+dx/2.  
  
  sum=sum+f(x)*dx  
enddo  
print*,sum  
stop  
end
```

```
function f(x)  
  f=x**2  
  return  
end
```

Programm 14 $\int_0^1 x^2 dx$ integraali arvutamine alamprogrammiga **quanc8**

```
external func  
a=0.  
b=1.  
abserr=0.0000  
relerr=0.0000001  
call quanc8(func,a,b,abserr,relerr,result,errest,nofun,flag)  
print *,result*4.,errest  
stop  
end  
function func(x)  
  func=sqrt(abs(1.-x*x))  
  return  
end
```

Parameetrite kirjelds

call quanc8(func,a,b,abserr,relerr,result,errest,nofun,flag)

sisend func – alamfunktsiooni nime (funktsiooni väärtuste arvutamiseks)
sisend a,b – integraali piirväärtused
sisend abserr- absoluutne viga
sisend relerr – suhtline viga
väljund result – integraali väärtus
väljund errest – integraali arvutamise täpsus ($result \pm errest$)
väljund flag – kui võrdub 0 siis kõik on korras

Programm 13 *sin,cos ja tan tabeli trükkimine*

```

pi=3.1415926536
print *," x[rad]      x[grad]      sin      cos      tan"
print *,"=====
do x=0.,pi/2.,0.01
    print '(5f12.4)',x,x*180/pi,sin(x),cos(x),tan(x)
enddo

stop
end

```

Programm 15 *π -arvu väärtuse arvutamine Monte-Carlo meetodil*

```

read *,n
nr=0
do i=1,n
    x=rand()
    y=rand()
    if ((x*x+y*y).le.1)then
        nr=nr+1
    endif
enddo
print*,4.*nr/n
stop
end

```

Programm 16 $\int_0^1 x^2 dx$ *integraali arvutamine Monte-Carlo meetodil*

```

b=1.
fb=f(b)
n=10000000
ns=0
do i=1,n
    xj=rand(0)*b
    yj=rand(0)*fb
    y=f(xj)
    if(y.ge.yj)then
        ns=ns+1
    endif
enddo
print *,ns*b*fb/n
stop
end

function f(x)
f=x*x
return
end

```

Programm 17 kahe vektorite vahelise nurga väärtuse arvutamine

```
real a(3),b(3),angle
open(10,file="test17.in")
read (10,*) a,b
call nurk(a,b,angle)
stop
end
subroutine nurk(a,b,angle)
pi=3.1415926536
ab=0.
ap=0.
bp=0.
do i=1,3
    ap=ap+a(i)**2
    bp=bp+b(i)**2
    ap=sqrt(ap)
    bp=sqrt(bp)
    ab=ab+a(i)*b(i)
enddo
angle=acos(ab/(ap*bp))*180./pi
return
end
```