

```

PROGRAM pret1
! oblicza przemieszczenia preta obciążonego osiowo
! MES: liniowe funkcje kształtu
! kodował:      M. Kuczma
! ostatnie zmiany: 05.11.01 (MSK)

```

```

IMPLICIT NONE
CHARACTER (LEN=80):: opis, dane, wyniki
INTEGER      :: i,k,n,m,i1,lelem,lssk,ne,ne1
REAL        :: c,d,dlugosc,EA0,he,xi,f0,alfa,beta
REAL        :: Ke(2,2), fe(2)
REAL, ALLOCATABLE :: a(:,:)

```

```

WRITE(*,*) 'PODAJ NAZWE ZBIORU DANYCH: '
READ(*,'(A15)') dane
OPEN(11, FILE=dane, STATUS='OLD', ACTION='READ')
WRITE(*,*) 'PODAJ NAZWE ZBIORU WYNIKOW: '
READ(*,'(A15)') wyniki
OPEN(12, FILE=wyniki, STATUS='REPLACE', ACTION='WRITE')

```

```

READ(11,'(A80)') opis ! nazwa zadania, identyfikacja
WRITE(12,'(A80)') opis ! nazwa zadania -> opis wynikow

```

```

READ(11,*) opis ! lelem = Liczba ELEMentow
READ(11,*) lelem
lssk=lelem+1 ! Liczba Stopni Swobody Konstrukcji
n=lssk; m=lssk+1 ! m=n+1; kolumna dla wektora obciazenia
ALLOCATE(a(n,m)) ! nadanie wym. mac. a i przydzielenie pamieci
READ(11,*) opis ! dlugosc preta
READ(11,*) dlugosc

```

```

he=dlugosc/lelem ! dlugosc elementow stala

```

```

READ(11,*) opis ! EA = modul Younga * pole przekroju preta
READ(11,*) EA0 ! stala dla calego preta
WRITE(12,*) '          DANE '
WRITE(12,'(A,F6.3,2x,A,I4,2x,A,E12.6,2x,A,E12.6)') 'dlugosc =', &
    dlugosc, ' lelem =', lelem, ' he =', he, ' EA0 =', EA0
WRITE(12,*) ' '
WRITE(12,*) '    obl. wg pret1.f90 '
WRITE(12,*) ' '

```

```

! obl. mac Ke, taka sama dla wszystkich elementow
Ke=macKe(EA0,he)
write(12,*) '    mac K^e, taka sama dla wszystkich elementow '
DO i=1,2
    write(12,'(I4,2X,2E16.6)') i, Ke(i,:)
END DO

```

```

! agregacja mac sztywnosci K dla calego preta
a=0. ! wyzerowanie mac. a
DO ne=1,lelem
    ne1=ne+1
    a(ne:ne1,ne:ne1)=a(ne:ne1,ne:ne1)+Ke(:,:)
END DO ! ne

```

```

READ(11,*) opis ! dane dot. obciazenia preta
READ(11,*) f0, alfa, beta
WRITE(12,*) ' '
WRITE(12,*) '    parametry OBCIAZENIA CIAGLEGO '
WRITE(12,'(3(2x,A,F8.4))') 'f0 =', f0, ' alfa =', alfa, ' beta =', beta
WRITE(12,*) ' '

```

```

! obl. mac. (wektora) obciazen wezlowych
DO ne=1,lelem
  xi=(ne-1)*he
  fe=wekfe(xi,he,f0,alfa,beta)
  WRITE(12,*) ' Wektor obc. wezlowych elementu e =', ne
  DO i=1,2
    write(12,'(I4,2X,2E16.6)') i, fe(i)
  END DO
  ne1=ne+1
  a(ne:ne1,m)=a(ne:ne1,m)+fe(:)
END DO ! ne

```

```

! uwzględnienie warunkow brzegowych
! przyjeto u(0)=0, u(l)=0 tj. konce preta nie przemieszczaja sie
  a(:,1)=0.;    a(1,:)=0.
  a(:,lssk)=0.; a(lssk,:)=0.
  a(1,1)=1.;    a(lssk,lssk)=1.
! koniec: otrzymano URL:  $K*u = f$ 

```

```

! rozwiazanie ukkladu rownan postaci
!  $a_{11}*x_1 + a_{12}*x_2 + \dots + a_{1n}*x_n = a_{1,n+1}, \dots, a_{1m}$ 
! .....
!  $a_{n1}*x_1 + a_{n2}*x_2 + \dots + a_{nn}*x_n = a_{n,n+1}, \dots, a_{nm}$ 
! metoda Gaussa-Jordana
! bez porzadkowania kolumn, eliminacja w porzadku wierszowym
! n = liczba niewiadomych (= lssk), m = n + 1

```

```

pta_i: DO i=1,n
  c=a(i,i)
  a(i,:)=a(i,:)/c
  i1=i+1
pta_k: DO k=1,n
  d=a(k,i)
  IF(k==i) CYCLE
  a(k,i1:m)=a(k,i1:m)-a(i,i1:m)*d
END DO pta_k
END DO pta_i

```

```

! koniec rozwiazywania ukkladu rownan

```

```

! wydruk rozwiazania u_i
WRITE(12,*) ' WYNIKI wg pret1.f90'
WRITE(12,*) ' i   x_i   u_i '
DO i=1,n !n=lssk
  xi=(i-1)*he
  WRITE(12,'(I4,2X,2E14.6)') i, xi, a(i,m) ! m=n+1
END DO

```

```

STOP

```

```

CONTAINS

```

```

FUNCTION macKe (AE,h)

```

```

! obl. mac sztywnosci elementu dwuwiezlowego, o liniowych funkcjach

```

```

! ksztaltu dla rozciaganego/sciskanego preta

```

```

! AE = sztywnosc przekroju elementu

```

```

! h = dlugosc elementu

```

```

! kodowal MSK, 14.10.01

```

```

IMPLICIT NONE

```

```

REAL, INTENT(IN) :: AE,h

```

```

REAL:: macKe(2,2), r

```

```

r=AE/h

```

```
macKe(1,:)= (/ r, -r /)
macKe(2,:)= (/ -r, r /)
RETURN
END FUNCTION macKe
```

```
FUNCTION wekfe(xi,h,f0,alfa,beta)
! obl. wektor obc. przywezlowych elementu dwuwezlowego,
! o liniowych funkcjach ksztaltu dla rozciaganego/sciskanego preta
! xi = wspolzedna lewego konca w ukladzie globalnym
! h = dlugosc elementu
! f0,alfa,beta = wspolczynniki funkcji obc.
! f(x) = f0 + alfa*x + beta*x^2 (w ukl. globalnym)
! kodowal: MSK, 14.10.01
IMPLICIT NONE
REAL, INTENT(IN) :: xi,h,f0,alfa,beta
REAL:: wekfe(2)
wekfe(1)=h*(6*beta*xi**2+4*beta*xi*h+beta*h**2+6*alfa*xi &
+2*alfa*h+6*f0)/12
wekfe(2)=h*(6*beta*xi**2+8*beta*xi*h+3*beta*h**2+6*alfa*xi &
+4*alfa*h+6*f0)/12
RETURN
END FUNCTION wekfe
```

```
END PROGRAM pret1
```