

```

function [KE,rtet,A]=matriz_es(ntot,iejes,NP,r,KL,RT)

%
% Programa que calcula la Matriz de Rigidez Espacial en Coordenadas
% de Piso, considerando tres grados de libertad por planta.
%
%          Por: Roberto Aguiar Falconi
%          Septiembre de 2011
%          Revisada para estructuras irregulares en Octubre 2012
%          CEINCI-ESPE
%
%-----
% [KE]=matriz_es(ntot,iejes,alt,r,KL,RT)
%-----
% iejes= # de ejes de columnas en direccion de analisis sismico.
% ntot  Numero total de pórticos
% NP    Numero de Pisos
% KL=   Matriz que contiene las matrices de rigidez lateral de todos los
%       porticos. Primero los de sentido X.
% r=    Vector que contiene la distancia del portico al centro de masa, de
%       cada uno de los porticos, con signo, positivo antihorario. Un
%       solo valor por pórtico
% RT=   Archivo de datos que contiene todos los vectores r, en todos los pisos.
%       Para estructuras regulares se colocara RT=r, Pero en estructuras irregulares
%       el valor de R se debe indicar piso por piso en la diagonal.
nx=iejes; ny=ntot-nx;
%Submatrices de rigidez: KEE,
Kxx=zeros(NP,NP);Kyy=zeros(NP,NP);Kteta=zeros(NP,NP);cero=zeros(NP,NP);
Kxt=zeros(NP,NP);Kyt=zeros(NP,NP);for k=1:NP;identidad(k,k)=1;end;
KE=zeros(3*NP,3*NP);

```

```
te=input('\n Tipo de estructura : Regular=1 Irregular=2 \n ');
```

```
if te==1
```

```
    for i=1:ntot
```

```
        for k=1:NP
```

```
            rtet(k,k)=r(i);
```

```
        end
```

```
        rteta=rtet*rtet;
```

```
        ji=NP*(i-1)+1;jf=NP*(i-1)+NP;
```

```
        if i<=nx
```

```
            Kxx=Kxx+KL(ji:jf,1:NP);Kxt=Kxt+KL(ji:jf,1:NP)*rtet;
```

```
            Kteta=Kteta+KL(ji:jf,1:NP)*rteta;
```

```
            A(ji:ji+NP-1,:)= [identidad cero rtet];
```

```
        else
```

```
            Kyy=Kyy+KL(ji:jf,1:NP);Kyt=Kyt+KL(ji:jf,1:NP)*rtet;
```

```
            Kteta=Kteta+KL(ji:jf,1:NP)*rteta;
```

```
            A(ji:ji+NP-1,:)= [cero identidad rtet];
```

```
        end
```

```
    end
```

```
elseif te==2;
```

```
rtet=zeros(NP,NP);
```

```
n=1;
```

```
for i=1:ntot;
```

```
    rtet=RT(n:i*NP,:);
```

```
    n=n+NP;
```

```
    rteta=rtet*rtet;
```

```
    ji=NP*(i-1)+1;jf=NP*(i-1)+NP;
```

```
if i<=nx
```

```
    Kxx=Kxx+KL(ji:jf,1:NP);
```

```
    Kxt=Kxt+KL(ji:jf,1:NP)*rtet;
```

```

Kteta=Kteta+KL(ji:jf,1:NP)*rteta;
A(ji:ji+NP-1,:)=[identidad cero rtet];
KE=KE+A(ji:ji+NP-1,:)'*KL(ji:jf,1:NP)*A(ji:ji+NP-1,:);
else
Kyy=Kyy+KL(ji:jf,1:NP);
Kyt=Kyt+KL(ji:jf,1:NP)*rtet;
Kteta=Kteta+KL(ji:jf,1:NP)*rteta;
A(ji:ji+NP-1,:)=[cero identidad rtet];
KE=KE+A(ji:ji+NP-1,:)'*KL(ji:jf,1:NP)*A(ji:ji+NP-1,:);
end
end
end
%Matriz de rigidez espacial con 3 grados de libertad por planta
if te==1;
KE=[Kxx cero Kxt;cero Kyy Kyt;Kxt Kyt Kteta];
end

```