

La falla en el material se presenta cuando la trayectoria de esfuerzos en la matriz se intercepta con la envolvente de resistencia que se encuentra dada por la Ecuación 2-1, como ya se demostró en la Ecuación 2-61 la pendiente en la trayectoria de esfuerzos en la matriz depende de las relaciones entre las rigideces del cemento y de la matriz y la pendiente de los esfuerzos externos, esto implica que ésta pendiente cambia a medida que se degrada el cemento mineral.

De la Ecuación 2-61 se puede concluir que en el estado intacto, es decir cuando el cemento no haya experimentado ningún daño ($L=0$) la pendiente de la trayectoria de esfuerzos en la matriz puede ser mayor que la pendiente impuesta en el ensayo, (3.0 ó PTEE), es por ello que entre más grande sea la relación entre los módulos de rigidez del cemento y la matriz, más rápido se alcanzará la envolvente de esfuerzos y más frágil será el material, esto es especialmente cierto cuando el valor de la energía umbral que determina el inicio del daño r_0 es alta. Esto puede conducir a colapsos del material cuando es sometido al ensayo edométrico situación que no ocurriría en un material no cementado.

La limitación más representativa de las dos aplicaciones desarrolladas es la cantidad de parámetros que deben ser ingresados, además como ya se advirtió existen combinaciones de parámetros que pueden conducir a fallas súbitas no representativas del material, incluso cuando se esté modelando el ensayo edométrico, esto también puede ocurrir en la modelación de ensayos triaxiales cuando por ejemplo se usen relaciones modulares cemento - matriz altas a bajas confinamientos o a bajas succiones.

Los parámetros asociados a la matriz pueden ser obtenidos a partir de la caracterización de una arcilla orientada a encontrar los parámetros del estado crítico, los parámetros relacionados con la variación de la compresibilidad en estado virgen generada por la succión deben ser obtenidos a partir de ensayos de compresión isotrópica a diferentes succiones tal y como lo sugiere Alonso et al. (1990), estos autores también proponen otras trayectorias de esfuerzos para la obtención de los parámetros restantes relacionados con la compresibilidad del material debido a la succión.

Particularmente se considera necesario que se realicen ensayos edométricos en donde se apliquen ciclos de humedecimiento-secado al material arcilloso (exclusivamente de ser posible) de manera similar a los realizados por Pousada en las arcillas de Madrid (1984), el objeto de estos ensayos es encontrar los parámetros que definen y permiten calcular las deformaciones plásticas en la macro-estructura debidas a las deformaciones elásticas en la micro-estructura, estos ensayos se deben realizar considerando varios valores para el confinamiento.

Los parámetros del cemento deben ser inferidos a partir del comportamiento global de la roca, solo la concentración del cemento puede ser inferida a partir de la caracterización del material aunque esta debe recurrir a técnicas como la porosimetría, las secciones delgadas, etc. Sobre el parámetro r_0 ya se hicieron las recomendaciones para tratar de inferir su valor, el valor r_1 corresponden a una tasa de daño que debe ser calibrada, los módulos de rigidez deben establecerse como una proporción de los módulos de la matriz arcillosa (G^b/G^M) teniendo en cuenta que las relaciones modulares altas inducen fragilidad en el material, conociendo el valor

APLICACIÓN DE UN MODELO CONSTITUTIVO A UNA ROCA LODOSA

del módulo de rigidez al corte se puede inferir el valor del módulo Bulk si se usa un valor de la relación de Poisson del cemento mineral tomada de la literatura y se emplean las relaciones entre las mencionadas propiedades que propone la teoría de la elasticidad lineal.

La caracterización ideal del cemento implicaría que este fuera separado de la matriz arcillosa para ser sometido a ensayos de laboratorio que determinen sus propiedades físicas y mecánicas, esta caracterización es bastante difícil de llevar a cabo, pues a pesar de lograr la separación del cemento, se deben afrontar problemas para la obtención de una muestra susceptible de ser ensayada. Por tal razón se debe recurrir a: caracterizaciones indirectas a partir del comportamiento global como la mencionada previamente para el parámetro r_0 , las formulaciones de la teoría de la elasticidad y los valores recomendados en la literatura que se deben seleccionar teniendo en cuenta la litología del cemento.

6.2 Recomendaciones

Se recomienda que se prosiga con el desarrollo de aplicaciones que modelen la resistencia de los materiales estructurados usando el modelo de Pinyol et al. (2007) y otros modelos propuestos en la literatura, se debe considerar otras trayectorias de esfuerzo siendo especialmente interesantes las que le impongan tracciones a la muestra.

Es importante que se estudien y se incorporen a las aplicaciones desarrolladas posteriormente nuevos criterios de falla que tengan en cuenta por ejemplo la curvatura de las envolventes de falla para grandes valores del esfuerzo de confinamiento.

Es necesario que se emprendan nuevas investigaciones en donde se desarrollen ensayos edométricos y triaxiales con succión controlada, de esta manera se podrán establecer criterios que calibren el modelo y las aplicaciones desarrolladas.

Dadas las dificultades relacionadas con la caracterización del material, sería importante que se estudiara la posibilidad de construir en el laboratorio un material cementado sintético para que este sea sometido a ensayos triaxiales bajo diferentes trayectorias de esfuerzos.

Sobre las aplicaciones desarrolladas se advierte que la selección de parámetros requerirá de ajustes de variables como: el parámetro estructural, la energía umbral del daño y la tasa de daño del cemento, esto se conseguirá repitiendo la modelación varias veces con valores diferentes de los parámetros mencionados hasta conseguir el comportamiento deseado

No todas las combinaciones de parámetros conducirán a modelaciones exitosas, como ya mencionó cuando existen grandes relaciones modulares (Valores altos de RG ó de RKb) se pueden provocar fallas súbitas del material durante los primeros incrementos de esfuerzo sobre todo cuando el esfuerzo de confinamiento es bajo, esta situación conlleva a ver sólo la parte elástica de la curvas esfuerzo-deformación.

Cuando se desee usar la aplicación BBMYBONDcd4.m con el fin de modelar un ciclo carga-descarga, se sugiere que el esfuerzo especificado para iniciar la descarga sea inferior al 80% del esfuerzo de falla, ya que valores superiores pueden causar inestabilidades de origen numérico en la aplicación que pueden causar el bloqueo de la computadora usada; lo anterior implica que se debe conocer previamente el valor del esfuerzo de falla para ello se debe ejecutar la aplicación sin llevar a cabo el ciclo mencionado.

Bibliografía

1. Alonso E. E., Gens A., Josa A. (1990). A constitutive model for partially saturated soil, *Géotechnique*, vol 40 (3), pp. 405-430.
2. Alonso E. E., Vaunat J., Gens A. (1999). Modeling the mechanical behaviour of expansive clays. *Engineering Geology*, vol 54, pp 173-183.
3. Alonso E.E. y I. Berdugo (2005). Expansive Behaviour of Sulphate-Bearing Clays. Proceedings of International Conference on Problematic Soils, Eastern Mediterranean University, Famagusta, N. Cyprus, 1-25.
4. Carol I., Rizzi E. & Willam K. (2001). On the formulation of anisotropic elastic degradation. 1. Theory based on a pseudologarithmic damage tensor rate. *Int. J. Solids Struct.*, 38: 491-528.
5. Cotecchia F. & Santaloia F. 2003. Compression behaviour of structurally complex marine clays. *Soft Ground Engineering in Coastal Areas; Proc. Nakase Memorial Symposium, Yokosuka, Japan*: 63-72.
6. Fernández A. L. y Santamarina J. C. (2001). Effect of cementation on the small-strain parameters of sands, *Canadian Geotechnical Journal* 38: 191–199 (2001), pp 191 – 199.
7. Gens A. & Nova R. (1993). Conceptual bases for a constitutive model for bonded soils and weak rocks. *Symposium on Geotechnical Engineering Hard Soils-Soft Rocks, Athens*, vol, 1, pp. 485-494.
8. Hoyos L (2008). Notas de clase del curso :” Modelling Constitutive Behavior of Partially Saturated Soils
9. Hsu, S.C. y P.P. Nelson (1993). Characterization of creataceous clay shales in north America. *Symp. Geotech. Engng. Hard Soils-Soft Rocks (Athens)* 1:139-146.
10. Kavvas M. & Amorosi A. (2000). A constitutive model for structured soils. *Géotechnique* vol. 50 (3), pp. 263-273.
11. Leroueil S. & Vaughan P.R. (1990). The general and congruent effects of structure in natural soils and weak rocks. *Géotechnique*, 40(3):467:488.
12. Montero, J. & Cortés, R. 1989. Clasificación Regional de Amenaza por Deslizamientos. *Memorias 1er. Simposio Suramericano de Deslizamientos SCG–CSMM, Paipa, Colombia*.

13. Montero, J. M. (2003). Notas de clase de Geología, Facultad de Ingeniería, Universidad Nacional de Colombia.
14. Montero, J.M. y Torres, M. C. (2007). Comportamiento geotécnico de lutitas en los Andes Colombianos, Facultad de Ingeniería, Universidad Nacional de Colombia. XIII Conferencia Panamericana de Mecánica de Suelos e Ingeniería Geotécnica, ISSMGE, Margarita, Venezuela.
15. Patiño, G. I. (2000). Caracterización geomecánica y comportamiento de las lutitas de la carretera La Vega–Villeta. Octubre de 2000, Trabajo de grado para optar al título de Magíster en Ingeniería – Geotecnia, Facultad de Ingeniería, Universidad Nacional de Colombia.
16. Pinyol N., Vaunat, J. & Alonso, E.E. (2007). A constitutive model for soft clayey rocks which includes weathering effects. Géotechnique Symposium in Print “Stiff Sedimentary Clays. Genesis and Engineering Behaviour”.
17. Torres Suárez M. C. (2008). Síntesis del Estado del Conocimiento sobre el Comportamiento Esfuerzo–Deformación y Degradabilidad de Rocas Lodosas. Congreso Colombiano de Geotecnia 2008.
18. Vaunat, J. & Gens, A. (2003). Bond degradation and irreversible strains in soft argillaceous rocks. In: Cullighan, P.J., Einstein, H.H. & Whittle, A. (Eds.) Proceedings of 12th Panamerican Conference on Soil Mechanics and Geotechnical Engineering, ISSMGE, vol. 1, pp. 479-484.
19. Wong, R.C.K. (1998). Swelling and softening behaviour of La Biche Shale, Canadian Geotechnical Journal 35, pp. 206-221.

Anexos:

Anexo 1: Código BBMYBONDcd4.

Anexo 2: Código EDOM_hs

```

%%Modelo constitutivo para rocas blandas: BBM + Cemento con carga y descarga%
%%Ingreso de las características iniciales de la muestra%
fprintf('\n Ingrese el volumen específico inicial de la macroestructura  \n\n');
VMini=input('Ingrese VMini= ');
fprintf('\n Ingrese el volumen específico inicial de la microestructura  \n\n');
Vmini=input('Ingrese Vmini= ');
fprintf('\n Ingrese la concentración del cemento  \n\n');
Cb=input('Ingrese Cb= ');
%Ingreso de los parámetros necesarios durante la etapa de consolidación%
%de la muestra%
if Cb==0;
    Xini=0;
    Kbini=0;
    Gbini=0;
    ro=1;
    rl=1;
    fprintf('\n Ingrese la pendiente de la URL de la macroestructura  \n\n');
    kappaM=input('Ingrese kappaM= ');
    fprintf('\n Ingrese la pendiente de la URL de la microestructura  \n\n');
    kappam=input('Ingrese kappam= ');
    fprintf('\n Ingrese el esfuerzo de preconsolidación en condición saturada  \n\n');
    PM0=input('Ingrese PM0= ');
    fprintf('\n Ingrese el valor del esfuerzo de referencia en MPa  \n\n');
    PMc=input('Ingrese PMc= ');
    fprintf('\n Ingrese el valor de la succión impuesta durante el ensayo en MPa \n\n');
    s=input('Ingrese s= ');
    fprintf('\n Ingrese el valor del esfuerzo de consolidación ISOTROPICA PMo en MPa
\n\n');
    fprintf('\n El modelo funciona para suelos normalmente consolidados o ligeramente
sobreconsolidados \n');
    Po=input('Ingrese Po= ');
    if PM0<Po;
        PM0=Po;
        sprintf('El nuevo valor del esfuerzo de preconsolidación es:%f\n',PM0);
    end
%Ingreso de los parámetros necesarios durante la etapa de falla%
fprintf('\n Ingrese la pendiente de la pendiente de la NCL de la macroestructura
condición saturada \n\n');
lo=input('Ingrese lo= ');
fprintf('\n Ingrese la tasa de cambio de la compresibilidad con la succión \n\n');
r=input('Ingrese r= ');
fprintf('\n Ingrese el valor de la compresibilidad cuando s tienda a infinito en
Mpa-1 \n\n');
beta=input('Ingrese beta= ');
fprintf('\n Ingrese el valor de la pendiente de la línea del estado crítico \n\n');
M=input('Ingrese M= ');
fprintf('\n Ingrese el valor de incremento de la "cohesión" con la succión \n\n');
ks=input('Ingrese ks= ');
fprintf('\n Ingrese el valor del módulo de rigidez al corte de la macroestructura en
MPa\n\n');
GM=input('Ingrese GM= ');

```

```

else
    fprintf('\n Ingrese la pendiente de la URL de la macroestructura \n\n');
    kappaM=input('Ingrese kappaM= ');
    fprintf('\n Ingrese la pendiente de la URL de la microestructura \n\n');
    kappam=input('Ingrese kappam= ');
    fprintf('\n Ingrese el esfuerzo de preconsolidación en condición saturada \n\n');
    PM0=input('Ingrese PM0= ');
    fprintf('\n Ingrese el valor del esfuerzo de referencia en MPa \n\n');
    PMc=input('Ingrese PMc= ');
    fprintf('\n Ingrese el valor de la variable de compatabilidad de deformaciones
\n\n');
    Xini=input('Ingrese Xini= ');
    fprintf('\n Ingrese el valor del módulo de Bulk inicial para el cemento en MPa
\n\n');
    Kbini=input('Ingrese Kbini= ');
    fprintf('\n Ingrese el valor del módulo de rigidez al corte inicial para el cemento
en MPa\n\n');
    Gbini=input('Ingrese Gbini= ');
    fprintf('\n Ingrese la energía elástica inicial del cemento en MPa \n\n');
    ro=input('Ingrese ro= ');
    fprintf('\n Ingrese la tasa de daño del cemento \n\n');
    rl=input('Ingrese rl= ');
    fprintf('\n Ingrese el valor de la succión impuesta durante el ensayo en MPa \n\n');
    s=input('Ingrese s= ');
    fprintf('\n Ingrese el valor del esfuerzo de consolidación ISOTROPICA PMo en MPa
\n\n');
    fprintf('\n El modelo funciona para suelos normalmente consolidados o ligeramente
sobreconsolidados \n');
    Po=input('Ingrese Po= ');
    if PM0<Po;
        PM0=Po;
        sprintf('El nuevo valor del esfuerzo de preconsolidación es:%f\n',PM0);
    end
    %Ingreso de los parámetros necesarios durante la etapa de falla%
    fprintf('\n Ingrese la pendiente de la pendiente de la NCL de la macroestructura
condición saturada \n\n');
    lo=input('Ingrese lo= ');
    fprintf('\n Ingrese la tasa de cambio de la compresibilidad con la succión \n\n');
    r=input('Ingrese r= ');
    fprintf('\n Ingrese el valor de la compresibilidad cuando s tienda a infinito en
Mpa-1 \n\n');
    beta=input('Ingrese beta= ');
    fprintf('\n Ingrese el valor de la pendiente de la línea del estado crítico \n\n');
    M=input('Ingrese M= ');
    fprintf('\n Ingrese el valor de incremento de la "cohesión" con la succión \n\n');
    ks=input('Ingrese ks= ');
    fprintf('\n Ingrese el valor del módulo de rígidez al corte de la macroestructura en
MPa\n\n');
    GM=input('Ingrese GM= ');
end
%%

```



```

%%Etapa de Consolidación Isotrópica%%
%Paso1: Cálculo de modulos Bulk iniciales y constantes%
KMini=(PMc*VMini)/kappaM;
Kmini=((PMc+s)*Vmini)/kappam;
KMextini=(KMini*Kmini)/(KMini+Kmini);
C1=Xini/(1+Cb*Xini);
RG1=Gbini/(3*GM);
RKbl=Kbini/KMini;
RKel=KMextini/KMini;
%Paso2: Cálculo del incremento en el PM y en el Pbo a partir de Pext%
PMo=Po/(1+(RKbl*RKel*Cb*C1^2));
Pbo=RKbl*C1*PMo;
%Paso3: Determinación de los volúmenes para el esfuerzo de consolidación PMo%
if PMo<PMc;
    VMo=VMini;
    Vmo=Vmini;
else
    VMo=VMini-kappaM*log(PMo/PMc);
    Vmo=Vmini-kappam*log(PMo/PMc);
end;
%Paso4: Determinación de las deformaciones por el esfuerzo de%
%consolidación%
DEMvo=(VMini-VMo)/VMini;
DEmvo=(Vmini-Vmo)/Vmini;
%Paso5: Determinación de la deformación externa por el esfuerzo de consolidación%
DEextvo=DEMvo+DEmvo;
%Paso6: Determinación de la deformación volumétrica del cemento por el%
%esfuerzo de preconsolidación%
if DEMvo<=0;
    DEbvo=Pbo/Kbini;
    DEextvo=DEbvo/(C1*RKel);
    DEMvo=DEbvo/Xini+Cb*DEbvo;
    DEmvo=DEextvo-DEMvo;
    VMo=VMini-DEMvo*VMini;
    Vmo=Vmini-DEmvo*Vmini;
else
    DEbvo=Xini*DEMvo/(1+Xini*Cb);
end;
%Paso7: Determinación del incremento en el esfuerzo equivalente P en el%
%cemento y el cálculo de la energía%
DPbo=Kbini*DEbvo;
Pbini=0;
Pbo=Pbini+DPbo;
DUbo=Pbo*DEbvo/2;
Ubini=0;
Ubo=Ubini+DUbo;
%Paso8: Determinación de la variable de daño L y los módulos del cemento%
if Cb==0
    Lo=0;
elseif log(Ubo/ro)/r1<0;
    Lo=0;

```

```

else
    Lo=log(Ubo/ro)/r1;
end
Xo=Xini*exp(-Lo/2);
Kbo=Kbini*exp(-Lo);
Gbo=Gbini*exp(-Lo);
C2=Xo/(1+Cb*Xo);
if PMo<PMc;
    KMo=(PMc*Vmo)/kappaM;
    Kmo=((PMc+s)*Vmo)/kappam;
    KMexto=(KMo*Kmo)/(KMo+Kmo);
else
    KMo=(PMo*Vmo)/kappaM;
    Kmo=((PMo+s)*Vmo)/kappam;
    KMexto=(KMo*Kmo)/(KMo+Kmo);
end;
RG2=Gbo/(3*GM);
RKb2=Kbo/KMo;
RKe2=KMexto/KMo;
pqext(1,1:4)=0;
KMm(1,1)=KMini;
KMm(1,2)=Kmini;
KMm(1,3)=KMextini;
const(1,1)=C1;
const(1,2)=RG1;
const(1,3)=RKb1;
const(1,4)=RKe1;
PM(1,1:4)=0;
pppo(1,1:3)=0;
Mbond(1,1)=Xini;
Mbond(1,2:9)=0;
Mbond(1,10)=0;
Mbond(1,11)=Kbini;
Mbond(1,12)=Gbini;
Mmvol(1,1)=VMini;
Mmvol(1,2)=Vmini;
Mmvol(1,3:4)=0;
Mdefor(1,1:12)=0;
envol(1,1)=M*ks*s;
pqext(2,1)=Po;
pqext(2,2)=0;
pqext(2,3)=DEextvo;
pqext(2,4)=0;
KMm(2,1)=KMo;
KMm(2,2)=Kmo;
KMm(2,3)=KMexto;
const(2,1)=C2;
const(2,2)=RG2;
const(2,3)=RKb2;
const(2,4)=RKe2;
PM(2,1:4)=0;

```

```

pqp(2,1)=PMo;
pqp(2,2:3)=0;
Mbond(2,1)=Xo;
Mbond(2,2)=Pbo;
Mbond(2,3)=0;
Mbond(2,4)=Mbond(1,4)+Mbond(2,2);
Mbond(2,5)=Mbond(1,5)+Mbond(2,3);
Mbond(2,6)=DUbo;
Mbond(2,7)=0;
Mbond(2,8)=Mbond(2,6)+Mbond(2,7);
Mbond(2,9)=Mbond(1,9)+Mbond(2,8);
Mbond(2,10)=Lo;
Mbond(2,11)=Kbo;
Mbond(2,12)=Gbo;
Mmv(2,1)=VMO;
Mmv(2,2)=Vmo;
Mmv(2,3:4)=0;
Mdefor(2,1)=DEMvo;
Mdefor(2,2)=DEmvo;
Mdefor(2,3)=DEextvo;
Mdefor(2,4)=0;
Mdefor(2,5)=DEMvo;
Mdefor(2,6)=DEmvo;
Mdefor(2,7)=Mdefor(2,5)+Mdefor(2,6);
Mdefor(2,8)=0;
Mdefor(2,9)=0;
Mdefor(2,10)=Mdefor(2,8)+Mdefor(2,9);
Mdefor(2,11)=DEbvo;
Mdefor(2,12)=0;
envol(2,1)=M*pqp(2,1)+M*ks*s;
%%

%%Etapa de falla%
%Pasol: Determinación de la compresibilidad para una succión s%
ls=lo*((1-r)*exp(-beta*s)+r);
%Paso2: Determinación del esfuerzo de preconsolidación para la succión s%
PMOB=PMc*(PM0/PMc)^((lo-kappaM)/(ls-kappaM));
%Paso3: Cálculo de coordenadas (pB,qB) del primer punto de cedencia B%
%Cálculo de las coordenadas de qB y pB%
gamma=1+RG2*C2*Cb*C2;
etha=1+RKb2*C2*Cb*C2*RKe2;
PTM=(3*etha)/gamma;
a=1+M^2/(PTM^2);
b=M^2*(2*PMo-PMOB+ks*s)/PTM;
c=M^2*(PMo+ks*s)*(PMo-PMOB);
qMB=(-b+((b^2)-4*a*c)^0.5)/(2*a);
pMB=PMo+qMB/PTM;
%Cálculo de los esfuerzos externos%
qextB=qMB+RG2*(C2^2)*Cb*qMB;
pextB=Po+qextB/3;
%Cálculo de volúmenes para el primer incremento%

```

```

if pMB<PMc;
    VMB=VMo;
    VmB=Vmo;
elseif PMo<PMc && pMB>PMc;
    VMB=VMo-kappaM*log(pMB/PMc);
    VmB=Vmo-kappam*log(pMB/PMc);
else
    VMB=VMo-kappaM*log(pMB/PMo);
    VmB=Vmo-kappam*log(pMB/PMo);
end;
%Paso5: Deformación cortante elástica primer incremento, aplica solamente%
%macroestructura%
DEMes=qMB/(3*GM);
%Paso6: Verificación de la condición de falla%
qMFB=(M*pMB+M*ks*s);
vc=3;
%Paso7: Vol para el esfuerzo de preconsolidación en B%
if PMoB<PMc;
    VMoB=VMini;
    VmoB=Vmini;
else
    if PMo>PMc
        VMoB=VMo-kappaM*log(PMoB/PMo);
        VmoB=Vmo-kappam*log(PMoB/PMo);
    else
        VMoB=VMo-kappaM*log(PMoB/PMc);
        VmoB=Vmo-kappam*log(PMoB/PMc);
    end
end
DEMvB=(VMo-VMoB)/VMo;
DEmvB=(Vmo-VmoB)/Vmo;
DEextvB=DEMvB+DEmvB;
%Paso8: Deformaciones, esfuerzos y energía en el cemento%
DEbvB=DEMvB*C2;
DEbsB=DEMes*C2;
DPbB=Kbo*DEbvB;
DqbB=Gbo*DEbsB;
pbB=Pbo+DPbB;
qo=0;
qbB=qo+DqbB;
UbpB=pbB*DEbvB;
UbqB=qbB*DEbsB;
UbtB=Ubo+UbpB+UbqB;
%Paso9: variable de daño, modulos y relaciones modulares%
if Cb==0;
    LB=0;
elseif log(UbtB/ro)/r1<0;
    LB=0;
else
    LB=log(UbtB/ro)/r1;
end

```

```

XB=Xo*exp(-LB/2);
KbB=Kbini*exp(-LB);
GbB=Gbini*exp(-LB);
C3=XB/(1+Cb*XB);
if pMB<PMc;
    KMB=(PMc*VMB)/kappaM;
    KmB=((PMc+s)*VmB)/kappam;
    KMextB=(KMB*KmB)/(KMB+KmB);
else
    KMB=(pMB*VMB)/kappaM;
    KmB=((pMB+s)*VmB)/kappam;
    KMextB=(KMB*KmB)/(KMB+KmB);
end;
RG3=GbB/(3*GM);
RKb3=KbB/KMo;
RKe3=KMextB/KMB;
pqpo(3,1)=pMB;
pqpo(3,2)=qMB;
pqpo(3,3)=PMoB;
envol(3,1)=M*pqpo(3,1)+M*ks*s;
pqext(3,1)=pextB;
pqext(3,2)=qextB;
KMm(3,1)=KMB;
KMm(3,2)=KmB;
KMm(3,3)=KMextB;
const(3,1)=C3;
const(3,2)=RG3;
const(3,3)=RKb3;
const(3,4)=RKe3;
PM(3,1)=gamma;
PM(3,2)=etha;
PM(3,3)=PTM;
PM(3,4)=qMB;
Mbond(3,1)=XB;
Mbond(3,2)=DPbB;
Mbond(3,3)=DqjbB;
Mbond(3,4)=Mbond(2,4)+Mbond(3,2);
Mbond(3,5)=Mbond(2,5)+Mbond(3,3);
Mbond(3,6)=UbpB;
Mbond(3,7)=UbqB;
Mbond(3,8)=UbtB;
Mbond(3,9)=UbtB;
Mbond(3,10)=LB;
Mbond(3,11)=KbB;
Mbond(3,12)=GbB;
Mmvol(3,1)=VMB;
Mmvol(3,2)=VmB;
Mmvol(3,3)=VMoB;
Mmvol(3,4)=VmoB;
Mdefor(3,1)=DEMvB;
Mdefor(3,2)=DEmvB;

```

```

Mdefor(3,3)=DEextvB;
Mdefor(3,4)=DEMes;
Mdefor(3,5)=Mdefor(3,1);
Mdefor(3,6)=Mdefor(3,2);
Mdefor(3,7)=Mdefor(3,5)+Mdefor(3,6);
Mdefor(3,8)=0;
Mdefor(3,9)=0;
Mdefor(3,10)=Mdefor(3,4)+Mdefor(3,9);
Mdefor(3,11)=DEbvB;
Mdefor(3,12)=DEbsB;
pqext(3,3)=pqext(2,3)+Mdefor(3,7);
pqext(3,4)=pqext(2,4)+Mdefor(3,10);
%Factor de corRección de Ohmaki%
alf=(M*(M-9)*(M-3))/(9*(6-M)*(1-kappaM/lo));

%%

%Paso10: Se aplica el i-ésimo incremento de esfuerzo externo%
fprintf('\n Presione 1 si desea hacer un ensayo con un ciclo descarga y recarga?
\n\n');
fprintf('\n Presione 2 si desea hacer un ensayo tradicional \n\n');
dc=input('1 ó 2: ');
if dc==1;
    fprintf('\n Ingrese el esfuerzo cortante equivalente q (en MPa) en el cual se
iniciará la descarga \n\n');
    fprintf('\n el valor de q debe ser superior a:\n');
    disp(qextB);
    qextd=input('Ingrese qextd= ');
    while qextd<qextB;
        fprintf('\n el valor de q debe ser superior a qextB \n\n');
        qextd=input('Ingrese qextd= ');
    end;
    ccd=0;
    veri=0;%verificación de valores%
    vc=vc+1;
    Dqext=0.01*qextB;
    pqext(vc,2)=pqext(vc-1,2)+Dqext;
    pqext(vc,1)=pqext(vc-1,1)+(Dqext/3);
    %Paso11: Cálculo de los i-ésimos esfuerzos en la matriz y en el cemento%
    PM(vc,4)=(pqext(vc,2)-pqpo(vc-1,2)-Mbond(vc-1,5)*Cb*const(vc-1,1))/(1+const(vc-
1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
    PM(vc,1)=(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
    PM(vc,2)=1+const(vc-1,3)*const(vc-1,1)*Cb*const(vc-1,1)*const(vc-1,4);
    PM(vc,3)=(3*PM(vc,2))/PM(vc,1);
    pqpo(vc,2)=pqpo(vc-1,2)+PM(vc,4);
    pqpo(vc,1)=pqpo(vc-1,1)+(PM(vc,4)/PM(vc,3));
    Mbond(vc,3)=const(vc-1,2)*const(vc-1,1)*PM(vc,4);
    Mbond(vc,2)=const(vc-1,3)*const(vc-1,1)*(PM(vc,4)/PM(vc,3));
    Mbond(vc,5)=Mbond(vc-1,5)+Mbond(vc,3);

```

```

Mbond(vc,4)=Mbond(vc-1,4)+Mbond(vc,2);
envol(vc,1)=M*pppo(vc,1)+M*ks*s;
while (pppo(vc,2)<envol(vc,1));
    %Pasol2: Esfuerzo de preconsolidación i-ésimo%
    pppo(vc,3)=pppo(vc,1)+pppo(vc,2)^2/(M^2*(pppo(vc,1)+ks*s));
    %Pasol3: Volúmenes correspondientes al esfuerzo de
    %preconsolidación%
    Mmvol(vc,3)=Mmvol(vc-1,3)-ls*log(pppo(vc,3)/pppo(vc-1,3));
    Mmvol(vc,4)=Mmvol(vc-1,4)-kappam*log(pppo(vc,3)/pppo(vc-1,3));
    %Pasol4: Volúmenes correspondientes al esfuerzo i-ésimo en la
    %matriz%
    if Mmvol(vc,3)+kappaM*log(pppo(vc,3)/pppo(vc,1))>Mmvol(vc-1,1);
        Mmvol(vc,1)=Mmvol(vc-1,1);
    else
        Mmvol(vc,1)=Mmvol(vc,3)+kappaM*log(pppo(vc,3)/pppo(vc,1));
    end
    if Mmvol(vc,4)+kappam*log(pppo(vc,3)/pppo(vc,1))>Mmvol(vc-1,2);
        Mmvol(vc,2)=Mmvol(vc-1,2);
    else
        Mmvol(vc,2)=Mmvol(vc,4)+kappam*log(pppo(vc,3)/pppo(vc,1));
    end
    %Pasol5: Incremento en las deformaciones elásticas%
    Mdefor(vc,1)=(pppo(vc,1)-pppo(vc-1,1))/KMm(vc-1,1);
    Mdefor(vc,2)=(pppo(vc,1)-pppo(vc-1,1))/KMm(vc-1,2);
    Mdefor(vc,3)=Mdefor(vc,1)+Mdefor(vc,2);
    Mdefor(vc,4)=(pppo(vc,2)-pppo(vc-1,2))/(3*GM);
    %Pasol6: Deformaciones totales volumétricas%
    Mdefor(vc,5)=(Mmvol(vc-1,1)-Mmvol(vc,1))/Mmvol(vc-1,1);
    Mdefor(vc,6)=(Mmvol(vc-1,2)-Mmvol(vc,2))/Mmvol(vc-1,2);
    Mdefor(vc,7)=Mdefor(vc,5)+Mdefor(vc,6);
    %si defor externas son ceros utilizar proceso alternativo%
    if Mdefor(vc,7)==0;
        Mdefor(vc,11)=Mbond(vc,2)/Mbond(vc-1,11);
        Mdefor(vc,12)=Mbond(vc,3)/Mbond(vc-1,12);
        Mdefor(vc,10)=Mdefor(vc,12)/const(vc-1,1);
        Mdefor(vc,4)=PM(vc,4)/(3*GM);
        if ccd==1 && pqext(vc,2)<(qextd-0.05*qextd);
            Mdefor(vc,9)=0;
            Mdefor(vc,8)=0;
        else
            Mdefor(vc,9)=Mdefor(vc,10)-Mdefor(vc,4);
            Mdefor(vc,8)=Mdefor(vc,9)/((2*alf*pppo(vc,2))/(M^2*(2*pppo(vc,1)+
ks*s-pppo(vc,3))));
        end
        Mdefor(vc,7)=Mdefor(vc,11)/(const(vc-1,1)*const(vc-1,4));
        Mdefor(vc,5)=Mdefor(vc,11)/const(vc-1,1);
        Mdefor(vc,6)=Mdefor(vc,7)-Mdefor(vc,5);
        Mdefor(vc,1)=Mdefor(vc,5)-Mdefor(vc,8);
        Mdefor(vc,2)=Mdefor(vc,6);
        Mdefor(vc,3)=Mdefor(vc,1)+Mdefor(vc,2);
        Mmvol(vc,1)=Mmvol(vc-1,1)-Mdefor(vc,5)*Mmvol(vc-1,1);

```

```

    Mmvol(vc,2)=Mmvol(vc-1,2)-Mdefor(vc,6)*Mmvol(vc-1,2);
else
    %Paso17: Deformaciones plásticas volumétricas%
    if Mdefor(vc,5)-Mdefor(vc,1)<0;
        Mdefor(vc,8)=0;
    elseif ccd==1 && pqext(vc,2)<(qextd-0.02*qextd);
        Mdefor(vc,8)=0;
    else
        Mdefor(vc,8)=Mdefor(vc,5)-Mdefor(vc,1);
    end
    %Paso18: Deformaciones plásticas cortantes%
    Mdefor(vc,9)=Mdefor(vc,8)*(2*alf*pppo(vc,2))/(M^2*(2*pppo(vc,1)+ks*s-
pppo(vc,3)));
    %Paso19: Deformación cortante total%
    Mdefor(vc,10)=Mdefor(vc,4)+Mdefor(vc,9);
    %Paso20: Deformaciones, esfuerzos y energía en el cemento%
    Mdefor(vc,11)=Mdefor(vc,5)*const(vc-1,1);
    Mdefor(vc,12)=Mdefor(vc,10)*const(vc-1,1);
end
Mbond(vc,6)=Mbond(vc,4)*Mdefor(vc,11);
Mbond(vc,7)=Mbond(vc,5)*Mdefor(vc,12);
Mbond(vc,8)=Mbond(vc,6)+Mbond(vc,7);
Mbond(vc,9)=Mbond(vc-1,9)+Mbond(vc,8);
%Paso21: Variable de daño, módulos y relaciones modulares%
if Cb==0;
    Mbond(vc,10)=0;
elseif log(Mbond(vc,9)/ro)/r1<0 && Mbond(vc-1,10)==0;
    Mbond(vc,10)=0;
else
    Mbond(vc,10)=log(Mbond(vc,9)/ro)/r1;
end;
Mbond(vc,1)=Mbond(1,1)*exp(-Mbond(vc,10)/2);
Mbond(vc,11)=Mbond(1,11)*exp(-Mbond(vc,10));
Mbond(vc,12)=Mbond(1,12)*exp(-Mbond(vc,10));
if pppo(vc,1)>PMc;
    KMm(vc,1)=pppo(vc,1)*Mmvol(vc,1)/kappaM;
    KMm(vc,2)=(pppo(vc,1)+s)*Mmvol(vc,2)/kappam;
    KMm(vc,3)=KMm(vc,1)*KMm(vc,2)/(KMm(vc,1)+KMm(vc,2));
else
    KMm(vc,1)=PMc*Mmvol(vc,1)/kappaM;
    KMm(vc,2)=(PMc+s)*Mmvol(vc,2)/kappam;
    KMm(vc,3)=(KMm(vc,1)*KMm(vc,2))/(KMm(vc,1)+KMm(vc,2));
end
const(vc,1)=Mbond(vc,1)/(1+Cb*Mbond(vc,1));
const(vc,2)=Mbond(vc,12)/(3*GM);
const(vc,3)=Mbond(vc,11)/KMm(vc,1);
const(vc,4)=KMm(vc,3)/KMm(vc,1);
%Paso22: Deformaciones externas%
pqext(vc,3)=pqext(vc-1,3)+Mdefor(vc,7);
pqext(vc,4)=pqext(vc-1,4)+Mdefor(vc,10);
%Se inicia la descarga si la condición se cumple%

```



```

if abs(pqext(vc,2)-qextd)<=0.01*qextd && ccd==0;
    vcd=vc;
    ccd=1;
    while pqext(vc,2)>=0;
        vc=vc+1;
        Dqextd=5*Dqext;
        %Esfuerzos externos%
        pqext(vc,2)=pqext(vc-1,2)-Dqextd;
        pqext(vc,1)=pqext(vc-1,1)-(Dqextd/3);
        %Esfuerzos en la matriz%
        PM(vc,4)=(pqext(vc,2)-(pqpo(vc-1,2)+Mbond(vc-1,5)*const(vc-1,1)*Cb))/
/(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
        PM(vc,1)=(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
        PM(vc,2)=1+const(vc-1,3)*const(vc-1,1)*Cb*const(vc-1,1)*const(vc-
1,4);

        PM(vc,3)=(3*PM(vc,2))/PM(vc,1);
        pqpo(vc,2)=pqpo(vc-1,2)+PM(vc,4);
        pqpo(vc,1)=pqpo(vc-1,1)+(PM(vc,4)/PM(vc,3));
        %Esfuerzos en el cemento%
        Mbond(vc,3)=const(vc-1,2)*const(vc-1,1)*PM(vc,4);
        Mbond(vc,2)=const(vc-1,3)*const(vc-1,1)*(PM(vc,4)/PM(vc,3));
        Mbond(vc,5)=Mbond(vc-1,5)+Mbond(vc,3);
        Mbond(vc,4)=Mbond(vc-1,4)+Mbond(vc,2);
        envol(vc,1)=M*pqpo(vc,1)+M*ks*s;
        if (pqpo(vc,2)>=envol(vc,1));
            disp('Se alcanzó la condición de falla');
            break;
        end;
        %Pasol2: Esfuerzo de preconsolidación i-ésimo%
        pqpo(vc,3)=pqpo(vc,1)+pqpo(vc,2)^2/(M^2*(pqpo(vc,1)+ks*s));
        %Pasol3: Volúmenes correspondientes al esfuerzo de%
        %preconsolidación%
        Mmvol(vc,3)=Mmvol(vc-1,3)-ls*log(pqpo(vc,3)/pqpo(vc-1,3));
        Mmvol(vc,4)=Mmvol(vc-1,4)-kappam*log(pqpo(vc,3)/pqpo(vc-1,3));
        %Pasol4: Volúmenes correspondientes al esfuerzo i-ésimo en la%
        %matriz%
        if Mmvol(vc,3)+kappaM*log(pqpo(vc,3)/pqpo(vc,1))<Mmvol(vc-1,1);
            Mmvol(vc,1)=Mmvol(vc-1,1);
        else
            Mmvol(vc,1)=Mmvol(vc,3)+kappaM*log(pqpo(vc,3)/pqpo(vc,1));
        end
        if Mmvol(vc,4)+kappam*log(pqpo(vc,3)/pqpo(vc,1))<Mmvol(vc-1,2);
            Mmvol(vc,2)=Mmvol(vc-1,2);
        else
            Mmvol(vc,2)=Mmvol(vc,4)+kappam*log(pqpo(vc,3)/pqpo(vc,1));
        end
        %Pasol5: Incremento en las deformaciones elásticas%
        Mdefor(vc,1)=(pqpo(vc,1)-pqpo(vc-1,1))/KMm(vc-1,1);
        Mdefor(vc,2)=(pqpo(vc,1)-pqpo(vc-1,1))/KMm(vc-1,2);
        Mdefor(vc,3)=Mdefor(vc,1)+Mdefor(vc,2);
        Mdefor(vc,4)=(pqpo(vc,2)-pqpo(vc-1,2))/(3*GM);

```

```

%Pasol6: Deformaciones totales volumétricas%
Mdefor(vc,5)=(Mmvol(vc-1,1)-Mmvol(vc,1))/Mmvol(vc-1,1);
Mdefor(vc,6)=(Mmvol(vc-1,2)-Mmvol(vc,2))/Mmvol(vc-1,2);
Mdefor(vc,7)=Mdefor(vc,5)+Mdefor(vc,6);
if Mdefor(vc,7)==0;
    Mdefor(vc,11)=Mbond(vc,2)/Mbond(vc-1,11);
    Mdefor(vc,12)=Mbond(vc,3)/Mbond(vc-1,12);
    Mdefor(vc,10)=Mdefor(vc,12)/const(vc-1,1);
    Mdefor(vc,4)=PM(vc,4)/(3*GM);
    Mdefor(vc,9)=Mdefor(vc,10)-Mdefor(vc,4);
    Mdefor(vc,8)=Mdefor(vc,9)/((2*alf*pppo(vc,2))/(M^2*(2*pppo(vc,1)
+ks*s-pppo(vc,3)))));
    Mdefor(vc,7)=Mdefor(vc,11)/(const(vc-1,1)*const(vc-1,4));
    Mdefor(vc,5)=Mdefor(vc,11)/const(vc-1,1);
    Mdefor(vc,6)=Mdefor(vc,7)-Mdefor(vc,5);
    Mdefor(vc,1)=Mdefor(vc,5)-Mdefor(vc,8);
    Mdefor(vc,2)=Mdefor(vc,6);
    Mdefor(vc,3)=Mdefor(vc,1)+Mdefor(vc,2);
    Mmvol(vc,1)=Mmvol(vc-1,1)-Mdefor(vc,5)*Mmvol(vc-1,1);
    Mmvol(vc,2)=Mmvol(vc-1,2)-Mdefor(vc,6)*Mmvol(vc-1,2);
else
    %Pasol7: Deformaciones plásticas volumétricas%
    if Mdefor(vc,5)-Mdefor(vc,1)<0;
        Mdefor(vc,8)=0;
    else
        Mdefor(vc,8)=Mdefor(vc,5)-Mdefor(vc,1);
    end
    %Pasol8: Deformaciones plásticas cortantes%
    Mdefor(vc,9)=Mdefor(vc,8)*(2*alf*pppo(vc,2))/(M^2*(2*pppo(vc,1)
+ks*s-pppo(vc,3)))));
    %Pasol9: Deformación cortante total%
    Mdefor(vc,10)=Mdefor(vc,4)+Mdefor(vc,9);
    %Paso20: Deformaciones, esfuerzos y energía en el cemento%
    Mdefor(vc,11)=Mdefor(vc,5)*const(vc-1,1);
    Mdefor(vc,12)=Mdefor(vc,10)*const(vc-1,1);
    veri=(~isreal(Mdefor));
    if veri==1;
        disp('Falla durante la descarga, disminuya qextd');
        break;
    end;
end
Mbond(vc,6)=Mbond(vc,4)*Mdefor(vc,11);
Mbond(vc,7)=Mbond(vc,5)*Mdefor(vc,12);
Mbond(vc,8)=Mbond(vc,6)+Mbond(vc,7);
Mbond(vc,9)=Mbond(vc-1,9)+Mbond(vc,8);
if Mbond(vc,9)<=0;
    Mbond(vc,9)=0;
end;
%Paso21: Variable de daño, módulos y relaciones modulares%
if Cb==0;
    Mbond(vc,10)=0;

```

```

elseif log(Mbond(vc,9)/ro)/r1<0 && Mbond(vc-1,10)==0;
    Mbond(vc,10)=0;
else
    Mbond(vc,10)=log(Mbond(vc,9)/ro)/r1;
    if Mbond(vc,10)<Mbond(vc-1,10);
        Mbond(vc,10)=Mbond(vc-1,10);
    end;
end;
veri=(~isreal(Mbond));
if veri==1;
    Cb=0;
    Mbond(vc,10)=0;
end;
Mbond(vc,1)=Mbond(1,1)*exp(-Mbond(vc,10)/2);
Mbond(vc,11)=Mbond(1,11)*exp(-Mbond(vc,10));
Mbond(vc,12)=Mbond(1,12)*exp(-Mbond(vc,10));
if pqpo(vc,1)>PMc;
    KMm(vc,1)=pqpo(vc,1)*Mmvol(vc,1)/kappaM;
    KMm(vc,2)=(pqpo(vc,1)+s)*Mmvol(vc,2)/kappam;
    KMm(vc,3)=KMm(vc,1)*KMm(vc,2)/(KMm(vc,1)+KMm(vc,2));
else
    KMm(vc,1)=PMc*Mmvol(vc,1)/kappaM;
    KMm(vc,2)=(PMc+s)*Mmvol(vc,2)/kappam;
    KMm(vc,3)=(KMm(vc,1)*KMm(vc,2))/(KMm(vc,1)+KMm(vc,2));
end
const(vc,1)=Mbond(vc,1)/(1+Cb*Mbond(vc,1));
const(vc,2)=Mbond(vc,12)/(3*GM);
const(vc,3)=Mbond(vc,11)/KMm(vc,1);
const(vc,4)=KMm(vc,3)/KMm(vc,1);
%Paso22: Deformaciones externas%
pqext(vc,3)=pqext(vc-1,3)+Mdefor(vc,7);
pqext(vc,4)=pqext(vc-1,4)+Mdefor(vc,10);
if abs((pqext(vc,1)-Po))<=0.05*Po;
    vcr=vc;
    break;
end;
end;
end;
end;
%Paso23: Se inicia la siguiente iteración cargando hasta la falla%
vc=vc+1;
%Esfuerzos externos%
pqext(vc,2)=pqext(vc-1,2)+Dqext;
pqext(vc,1)=pqext(vc-1,1)+(Dqext/3);
%Esfuerzos en la matriz%
PM(vc,4)=(pqext(vc,2)-(pqpo(vc-1,2)+Mbond(vc-1,5)*const(vc-1,1)*Cb))/
(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
PM(vc,1)=(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
PM(vc,2)=1+const(vc-1,3)*const(vc-1,1)*Cb*const(vc-1,1)*const(vc-1,4);
PM(vc,3)=(3*PM(vc,2))/PM(vc,1);
pqpo(vc,2)=pqpo(vc-1,2)+PM(vc,4);
pqpo(vc,1)=pqpo(vc-1,1)+(PM(vc,4)/PM(vc,3));

```

```

%Esfuerzos en el cemento%
Mbond(vc,3)=const(vc-1,2)*const(vc-1,1)*PM(vc,4);
Mbond(vc,2)=const(vc-1,3)*const(vc-1,1)*(PM(vc,4)/PM(vc,3));
Mbond(vc,5)=Mbond(vc-1,5)+Mbond(vc,3);
Mbond(vc,4)=Mbond(vc-1,4)+Mbond(vc,2);
envol(vc,1)=M*pppo(vc,1)+M*ks*s;
if (pppo(vc,2)>=envol(vc,1));
    disp('Se alcanzó la condición de falla');
    break;
end;
if veri==1;
    break;
end
end;

elseif dc==2;
    vc=vc+1;
    Dqext=0.01*qextB;
    pqext(vc,2)=pqext(vc-1,2)+Dqext;
    pqext(vc,1)=pqext(vc-1,1)+(Dqext/3);
    %Pasol1: Cálculo de los i-ésimos esfuerzos en la matriz y en el cemento%
    PM(vc,4)=(pqext(vc,2)-pppo(vc-1,2)-Mbond(vc-1,5)*Cb*const(vc-1,1))/(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
    PM(vc,1)=(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
    PM(vc,2)=1+const(vc-1,3)*const(vc-1,1)*Cb*const(vc-1,1)*const(vc-1,4);
    PM(vc,3)=(3*PM(vc,2))/PM(vc,1);
    pppo(vc,2)=pppo(vc-1,2)+PM(vc,4);
    pppo(vc,1)=pppo(vc-1,1)+(PM(vc,4)/PM(vc,3));
    Mbond(vc,3)=const(vc-1,2)*const(vc-1,1)*PM(vc,4);
    Mbond(vc,2)=const(vc-1,3)*const(vc-1,1)*(PM(vc,4)/PM(vc,3));
    Mbond(vc,5)=Mbond(vc-1,5)+Mbond(vc,3);
    Mbond(vc,4)=Mbond(vc-1,4)+Mbond(vc,2);
    envol(vc,1)=M*pppo(vc,1)+M*ks*s;
    while (pppo(vc,2)<envol(vc,1));
        %Pasol2: Esfuerzo de preconsolidación i-ésimo%
        pppo(vc,3)=pppo(vc,1)+pppo(vc,2)^2/(M^2*(pppo(vc,1)+ks*s));
        %Pasol3: Volúmenes correspondientes al esfuerzo de
        %preconsolidación%
        Mmvol(vc,3)=Mmvol(vc-1,3)-ls*log(pppo(vc,3)/pppo(vc-1,3));
        Mmvol(vc,4)=Mmvol(vc-1,4)-kappam*log(pppo(vc,3)/pppo(vc-1,3));
        %Pasol4: Volúmenes correspondientes al esfuerzo i-ésimo en la
        %matriz%
        if Mmvol(vc,3)+kappaM*log(pppo(vc,3)/pppo(vc,1))>Mmvol(vc-1,1);
            Mmvol(vc,1)=Mmvol(vc-1,1);
        else
            Mmvol(vc,1)=Mmvol(vc,3)+kappaM*log(pppo(vc,3)/pppo(vc,1));
        end
        if Mmvol(vc,4)+kappam*log(pppo(vc,3)/pppo(vc,1))>Mmvol(vc-1,2);
            Mmvol(vc,2)=Mmvol(vc-1,2);
        else
            Mmvol(vc,2)=Mmvol(vc,4)+kappam*log(pppo(vc,3)/pppo(vc,1));
        end
    end
end

```

```

end
%Paso15: Incremento en las deformaciones elásticas%
Mdefor(vc,1)=(pqpo(vc,1)-pqpo(vc-1,1))/KMm(vc-1,1);
Mdefor(vc,2)=(pqpo(vc,1)-pqpo(vc-1,1))/KMm(vc-1,2);
Mdefor(vc,3)=Mdefor(vc,1)+Mdefor(vc,2);
Mdefor(vc,4)=(pqpo(vc,2)-pqpo(vc-1,2))/(3*GM);
%Paso16: Deformaciones totales volumétricas%
Mdefor(vc,5)=(Mmvol(vc-1,1)-Mmvol(vc,1))/Mmvol(vc-1,1);
Mdefor(vc,6)=(Mmvol(vc-1,2)-Mmvol(vc,2))/Mmvol(vc-1,2);
Mdefor(vc,7)=Mdefor(vc,5)+Mdefor(vc,6);
%si defor externas son ceros utilizar proceso alternativo%
if Mdefor(vc,7)==0;
    Mdefor(vc,11)=Mbond(vc,2)/Mbond(vc-1,11);
    Mdefor(vc,12)=Mbond(vc,3)/Mbond(vc-1,12);
    Mdefor(vc,10)=Mdefor(vc,12)/const(vc-1,1);
    Mdefor(vc,4)=PM(vc,4)/(3*GM);
    Mdefor(vc,9)=Mdefor(vc,10)-Mdefor(vc,4);
    Mdefor(vc,8)=Mdefor(vc,9)/((2*alf*pqpo(vc,2))/(M^2*(2*pqpo(vc,1)+ks*s-
pqpo(vc,3)))));
    Mdefor(vc,7)=Mdefor(vc,11)/(const(vc-1,1)*const(vc-1,4));
    Mdefor(vc,5)=Mdefor(vc,11)/const(vc-1,1);
    Mdefor(vc,6)=Mdefor(vc,7)-Mdefor(vc,5);
    Mdefor(vc,1)=Mdefor(vc,5)-Mdefor(vc,8);
    Mdefor(vc,2)=Mdefor(vc,6);
    Mdefor(vc,3)=Mdefor(vc,1)+Mdefor(vc,2);
    Mmvol(vc,1)=Mmvol(vc-1,1)-Mdefor(vc,5)*Mmvol(vc-1,1);
    Mmvol(vc,2)=Mmvol(vc-1,2)-Mdefor(vc,6)*Mmvol(vc-1,2);
else
    %Paso17: Deformaciones plásticas volumétricas%
    if Mdefor(vc,5)-Mdefor(vc,1)<0;
        Mdefor(vc,8)=0;
    else
        Mdefor(vc,8)=Mdefor(vc,5)-Mdefor(vc,1);
    end
    %Paso18: Deformaciones plásticas cortantes%
    Mdefor(vc,9)=Mdefor(vc,8)*(2*alf*pqpo(vc,2))/(M^2*(2*pqpo(vc,1)+ks*s-
pqpo(vc,3)))));
    %Paso19: Deformación cortante total%
    Mdefor(vc,10)=Mdefor(vc,4)+Mdefor(vc,9);
    %Paso20: Deformaciones, esfuerzos y energía en el cemento%
    Mdefor(vc,11)=Mdefor(vc,5)*const(vc-1,1);
    Mdefor(vc,12)=Mdefor(vc,10)*const(vc-1,1);
end
Mbond(vc,6)=Mbond(vc,4)*Mdefor(vc,11);
Mbond(vc,7)=Mbond(vc,5)*Mdefor(vc,12);
Mbond(vc,8)=Mbond(vc,6)+Mbond(vc,7);
Mbond(vc,9)=Mbond(vc-1,9)+Mbond(vc,8);
%Paso21: Variable de daño, módulos y relaciones modulares%
if Cb==0;
    Mbond(vc,10)=0;
elseif log(Mbond(vc,9)/ro)/r1<0;

```

```

    Mbond(vc,10)=0;
else
    Mbond(vc,10)=log(Mbond(vc,9)/ro)/r1;
end;
Mbond(vc,1)=Mbond(1,1)*exp(-Mbond(vc,10)/2);
Mbond(vc,11)=Mbond(1,11)*exp(-Mbond(vc,10));
Mbond(vc,12)=Mbond(1,12)*exp(-Mbond(vc,10));
if pqpo(vc,1)>PMc;
    KMm(vc,1)=pqpo(vc,1)*Mmvol(vc,1)/kappaM;
    KMm(vc,2)=(pqpo(vc,1)+s)*Mmvol(vc,2)/kappam;
    KMm(vc,3)=KMm(vc,1)*KMm(vc,2)/(KMm(vc,1)+KMm(vc,2));
else
    KMm(vc,1)=PMc*Mmvol(vc,1)/kappaM;
    KMm(vc,2)=(PMc+s)*Mmvol(vc,2)/kappam;
    KMm(vc,3)=(KMm(vc,1)*KMm(vc,2))/(KMm(vc,1)+KMm(vc,2));
end
const(vc,1)=Mbond(vc,1)/(1+Cb*Mbond(vc,1));
const(vc,2)=Mbond(vc,12)/(3*GM);
const(vc,3)=Mbond(vc,11)/KMm(vc,1);
const(vc,4)=KMm(vc,3)/KMm(vc,1);
%Paso22: Deformaciones externas%
pqext(vc,3)=pqext(vc-1,3)+Mdefor(vc,7);
pqext(vc,4)=pqext(vc-1,4)+Mdefor(vc,10);
%Paso23: Se inicia la siguiente iteración%
vc=vc+1;
%Esfuerzos externos%
pqext(vc,2)=pqext(vc-1,2)+Dqext;
pqext(vc,1)=pqext(vc-1,1)+(Dqext/3);
%Esfuerzos en la matriz%
PM(vc,4)=(pqext(vc,2)-(pqpo(vc-1,2)+Mbond(vc-1,5)*const(vc-1,1)*Cb))/
(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
PM(vc,1)=(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
PM(vc,2)=1+const(vc-1,3)*const(vc-1,1)*Cb*const(vc-1,1)*const(vc-1,4);
PM(vc,3)=(3*PM(vc,2))/PM(vc,1);
pqpo(vc,2)=pqpo(vc-1,2)+PM(vc,4);
pqpo(vc,1)=pqpo(vc-1,1)+(PM(vc,4)/PM(vc,3));
Mbond(vc,3)=const(vc-1,2)*const(vc-1,1)*PM(vc,4);
Mbond(vc,2)=const(vc-1,3)*const(vc-1,1)*(PM(vc,4)/PM(vc,3));
Mbond(vc,5)=Mbond(vc-1,5)+Mbond(vc,3);
Mbond(vc,4)=Mbond(vc-1,4)+Mbond(vc,2);
envol(vc,1)=M*pqpo(vc,1)+M*ks*s;
if (pqpo(vc,2)>=envol(vc,1));
    disp('Se alcanzó la condición de falla');
    break;
end;
end;
end;
%Gráficas de los resultados%

```

```
%Gráfica de la trayectoria de esfuerzos%
subplot(2,2,1);
plot(pqext(2:vc,1),pqext(2:vc,2));
title('Trayectoria de Esfuerzos externos');
xlabel('Pext (MPa)');
ylabel('qext (Mpa)');
%Gráfica del volumen específico contra P en la matriz%
subplot(2,2,2);
plot(pqpo(2:vc-1,1),Mmvol(2:vc-1,1));
title('Curva de Compresibilidad de la macroestructura');
xlabel('PM (MPa)');
ylabel('vM=1+eM');
%Gráfica de qext vs la deformación cortante total%
subplot(2,2,3);
plot(pqext(2:vc-1,4),pqext(2:vc-1,2));
title('qext vs Deformación cortante total');
xlabel('Est');
ylabel('qext (Mpa)');
%Gráfica del volumen específico macro vs la deformación cortante total%
subplot(2,2,4);
plot(pqext(2:vc-1,4),Mmvol(2:vc-1,1));
title('v vs Deformación cortante total');
xlabel('Est');
ylabel('vM=1+eM');
```

```

%EDOM_hs: Simulación de ensayos edométricos en rocas sometidas a ciclos%
% de humedecimiento secado%
%%Ingreso de las características iniciales de la muestra%%
fprintf('\n Ingrese el volumen específico inicial de la macroestructura \n\n');
VMini=input('Ingrese VMini= ');
fprintf('\n Ingrese el volumen específico inicial de la microestructura \n\n');
Vmini=input('Ingrese Vmini= ');
fprintf('\n Ingrese la concentración del cemento \n\n');
Cb=input('Ingrese Cb= ');
%Ingreso de los parámetros necesarios durante la etapa de consolidación%
%de la muestra%
if Cb==0;
    Xini=0;
    Kbini=0;
    Gbini=0;
    ro=1;
    rl=1;
    %ingreso de los parámetros de la matriz%
    fprintf('\n Ingrese la pendiente de la URL de la macroestructura \n\n');
    kappaM=input('Ingrese kappaM= ');
    fprintf('\n Ingrese la pendiente de la URL de la microestructura \n\n');
    kappam=input('Ingrese kappam= ');
    fprintf('\n Ingrese la pendiente de la URL para succión \n\n');
    kappas=input('Ingrese kappas= ');
    fprintf('\n Ingrese la pendiente de la pendiente de la NCL de la macroestructura en
condición saturada \n\n');
    lo=input('Ingrese lo= ');
    fprintf('\n Ingrese el valor del módulo de rigidez al corte de la macroestructura en
MPa\n\n');
    GM=input('Ingrese GM= ');
    fprintf('\n Ingrese el valor de la pendiente de la línea del estado crítico \n\n');
    M=input('Ingrese M= ');
    fprintf('\n Ingrese el esfuerzo de preconsolidación neto en condición saturada
\n\n');
    PM0=input('Ingrese PM0= ');
    %Parámetros relacionados con el cambio de rigidez por succión%
    fprintf('\n Ingrese el valor del esfuerzo de referencia neta en MPa \n\n');
    PMc=input('Ingrese PMc= ');
    fprintf('\n Ingrese la tasa de cambio de la compresibilidad con la succión \n\n');
    r=input('Ingrese r= ');
    fprintf('\n Ingrese el valor de la compresibilidad cuando s tienda a infinito en
Mpa-1 \n\n');
    beta=input('Ingrese beta= ');
    fprintf('\n Ingrese el valor de incremento de la "cohesión" con la succión \n\n');
    ks=input('Ingrese ks= ');
    %Valores relacionados con el ensayo%
    fprintf('\n Ingrese el valor de la succión inicial impuesta durante el ensayo en MPa
\n\n');
    s=input('Ingrese s= ');
    fprintf('\n Ingrese el valor del esfuerzo vertical neto inicial en MPa \n\n');
    sigmavo=input('Ingrese sigmavo= ');

```



```

fprintf('\n Ingrese el valor del esfuerzo vertical neto final en MPa \n\n');
sigmavf=input('Ingrese sigmavf= ');
fprintf('\n Ingrese el valor del coeficiente de presión de tierras en reposo \n\n');
Ko=input('Ingrese Ko= ');
else
fprintf('\n Ingrese la pendiente de la URL de la macroestructura \n\n');
kappaM=input('Ingrese kappaM= ');
fprintf('\n Ingrese la pendiente de la URL de la microestructura \n\n');
kappam=input('Ingrese kappam= ');
fprintf('\n Ingrese la pendiente de la URL para succión \n\n');
kappas=input('Ingrese kappas= ');
fprintf('\n Ingrese la pendiente de la pendiente de la NCL de la macroestructura
condición saturada \n\n');
lo=input('Ingrese lo= ');
fprintf('\n Ingrese el valor del módulo de rigidez al corte de la macroestructura en
MPa\n\n');
GM=input('Ingrese GM= ');
fprintf('\n Ingrese el valor de la pendiente de la línea del estado crítico \n\n');
M=input('Ingrese M= ');
fprintf('\n Ingrese el esfuerzo de preconsolidación neto en condición saturada
\n\n');
PM0=input('Ingrese PM0= ');
%Parámetros relacionados con el cambio de rigidez por succión%
fprintf('\n Ingrese el valor del esfuerzo de referencia neta en MPa \n\n');
PMc=input('Ingrese PMc= ');
fprintf('\n Ingrese la tasa de cambio de la compresibilidad con la succión \n\n');
r=input('Ingrese r= ');
fprintf('\n Ingrese el valor de la compresibilidad cuando s tienda a infinito en
Mpa-1 \n\n');
beta=input('Ingrese beta= ');
fprintf('\n Ingrese el valor de incremento de la "cohesión" con la succión \n\n');
ks=input('Ingrese ks= ');
%Valor del parámetro estructural%
fprintf('\n Ingrese el valor de la variable de compatibilidad de deformaciones
\n\n');
Xini=input('Ingrese Xini= ');
%Parámetros relaciondos con el cemento%
fprintf('\n Ingrese el valor del módulo de Bulk inicial para el cemento en MPa
\n\n');
Kbini=input('Ingrese Kbini= ');
fprintf('\n Ingrese el valor del módulo de rigidez al corte inicial para el cemento
en MPa\n\n');
Gbini=input('Ingrese Gbini= ');
fprintf('\n Ingrese la energía elástica inicial del cemento en MPa \n\n');
ro=input('Ingrese ro= ');
fprintf('\n Ingrese la tasa de daño del cemento \n\n');
r1=input('Ingrese r1= ');
%Valores relacionados con el ensayo%
fprintf('\n Ingrese el valor de la succión inicial impuesta durante el ensayo en MPa
\n\n');
s=input('Ingrese s= ');

```

```

fprintf('\n Ingrese el valor del esfuerzo vertical inicial en MPa \n\n');
sigmavo=input('Ingrese sigmavo= ');
fprintf('\n Ingrese el valor del esfuerzo vertical neto final en MPa \n\n');
sigmavf=input('Ingrese sigmavf= ');
fprintf('\n Ingrese el valor del coeficiente de presión de tierras en reposo \n\n');
Ko=input('Ingrese Ko= ');
end

%%
%%PRIMER ENSAYO EDOMETRICO. MATERIAL "INTACTO"%%
%%Aplicación del esfuerzo inicial%%
%Paso1: Cálculo de esfuerzo equivalente inicial modulos Bulk iniciales y constantes%
Patm=0.1;
Po=sigmavo*(1+2*Ko)/3;%nuevo%
qo=sigmavo*(1-Ko);%nuevo%
KMini=(PMc*VMini)/kappaM;
Kmini=((PMc)*Vmini)/kappam;
KMextini=(KMini*Kmini)/(KMini+Kmini);
Cl=Xini/(1+Cb*Xini);
RGl=Gbini/(3*GM);
RKbl=Kbini/KMini;
RKel=KMextini/KMini;
%Paso2: Cálculo del incremento en la matriz y cemento a partir de esf externos%
PMo=Po/(1+(RKbl*RKel*Cb*Cl^2));
Pbo=RKbl*Cl*PMo;
qMo=qo/(1+(RGl*Cb*Cl^2));%n%
qbo=RGl*Cl*qMo;%n%
%Paso3: Determinación de los volúmenes para el esfuerzo inicial PMo%
if PMo<PMc;
    VMo=VMini;
    Vmo=Vmini;
else
    VMo=VMini-kappaM*log(PMo/PMc);
    Vmo=Vmini-kappam*log(PMo/PMc);
end;
%Paso4: Determinación de las deformaciones por el esfuerzo inicial%
DEMvo=(VMini-VMo)/VMini;
DEmvo=(Vmini-Vmo)/Vmini;
DEMeso=qMo/(3*GM);%n%
%Paso5: Determinación de la deformación externa por el esfuerzo inicial%
DEextvo=DEMvo+DEmvo;
%Paso6: Determinación de la deformación volumétrica del cemento por el
%esfuerzo inicial%
if DEMvo<=0;
    DEbvo=Pbo/Kbini;
    DEbso=qbo/Gbini;%n%
    DETotso=DEbso/Cl;%n%
    DEMeso=DETotso;%n%
    DEextvo=DEbvo/(Cl*RKel);
    DEMvo=DEbvo/Xini+Cb*DEbvo;
    DEmvo=DEextvo-DEMvo;

```

```

Vmo=VMini-DEMvo*VMini;
Vmo=Vmini-DEMvo*Vmini;
else
    DEbvo=Xini*DEMvo/(1+Xini*Cb);
    DEbso=C1*DEMeso;%n%
    DETotso=DEMeso;%n%
end;
%Paso7: Determinación del incremento en el esfuerzo equivalente P en el%
%cemento y el cálculo de la energía%
DPbo=Kbini*DEbvo;
Pbini=0;
Pbo=Pbini+DPbo;
Dqbo=Gbini*DEbso;%n%
qbini=0;%n%
qbo=qbini+Dqbo;%n%
DUbvollo=Pbo*DEbvo/2;%n%
DUbso=qbo*DEbso/2;%n%
DUbo=DUbvollo+DUbso;
Ubini=0;
Ubo=Ubini+DUbo;
%Paso8: Determinación de la variable de daño L y los módulos del cemento%
if Cb==0
    Lo=0;
elseif log(Ubo/ro)/r1<0;
    Lo=0;
else
    Lo=log(Ubo/ro)/r1;
end
Xo=Xini*exp(-Lo/2);
Kbo=Kbini*exp(-Lo);
Gbo=Gbini*exp(-Lo);
C2=Xo/(1+Cb*Xo);
if PMo<PMc;
    KMo=(PMc*VMo)/kappaM;
    Kmo=((PMc)*Vmo)/kappam;
    KMexto=(KMo*Kmo)/(KMo+Kmo);
else
    KMo=(PMo*VMo)/kappaM;
    Kmo=((PMo)*Vmo)/kappam;
    KMexto=(KMo*Kmo)/(KMo+Kmo);
end;
RG2=Gbo/(3*GM);
RKb2=Kbo/KMo;
RKe2=KMexto/KMo;
%Paso9: Determinación de las deformaciones por la succión inicial%
Kso=((Patm)*VMo)/kappas;
DEMvo=DEMvo+(s/Kso);
DEmvo=DEmvo+(s/Kmo);
DEextvo=DEextvo+(s/Kso)+(s/Kmo);
VMo=VMo-kappas*log((s+Patm)/Patm);
Vmo=Vmo-kappam*log((s+PMo)/PMo);

```

```

%Paso10: Determinación de las deformaciones o esfuerzos en el cemento%
DEbvo2=C2*(s/Kso+s/Kmo);
DPbo2=Kbini*DEbvo2;
%Paso11: Determinación de la nueva energía de deformación%
DUbvolo2=DPbo2*DEbvo2;
DUbo=DUbo+DUbvolo2;
Ubo=Ubo+DUbo;
if Cb==0
    Lo2=0;
elseif log(Ubo/ro)/r1<0;
    Lo2=0;
else
    Lo2=log(Ubo/ro)/r1;
end
Xo2=Xini*exp(-Lo2/2);
Kbo2=Kbini*exp(-Lo2);
Gbo2=Gbini*exp(-Lo2);
C2=Xo2/(1+Cb*Xo2);
if PMo<PMc;
    KMo=(PMc*Vmo)/kappaM;
    Kmo=((PMc+s)*Vmo)/kappam;
    KMexto=(KMo*Kmo)/(KMo+Kmo);
else
    KMo=(PMo*Vmo)/kappaM;
    Kmo=((PMo+s)*Vmo)/kappam;
    KMexto=(KMo*Kmo)/(KMo+Kmo);
end;
RG22=Gbo2/(3*GM);
RKb22=Kbo2/KMo;
RKe22=KMexto/KMo;
%Asignaciones a las matrices%
pqext(1,1:5)=0;
KMm(1,1)=KMini;
KMm(1,2)=Kmini;
KMm(1,3)=KMextini;
const(1,1)=C1;
const(1,2)=RG1;
const(1,3)=RKbl;
const(1,4)=RKel;
PM(1,1:4)=0;
pppo(1,1:3)=0;
Mbond(1,1)=Xini;
Mbond(1,2:9)=0;
Mbond(1,10)=0;
Mbond(1,11)=Kbini;
Mbond(1,12)=Gbini;
Mmvol(1,1)=VMini;
Mmvol(1,2)=Vmini;
Mmvol(1,3:4)=0;
Mdefor(1,1:12)=0;
envol(1,1)=M*ks*s;

```

```

pqext(2,1)=Po;
pqext(2,2)=qo;
pqext(2,3)=DEextvo;
pqext(2,4)=DEMeso;
KMm(2,1)=KMo;
KMm(2,2)=Kmo;
KMm(2,3)=KMexto;
const(2,1)=C2;
const(2,2)=RG22;
const(2,3)=RKb22;
const(2,4)=RKe22;
PM(2,1:4)=0;
pqpo(2,1)=PMo;
pqpo(2,2)=qMo;
pqpo(2,3)=0;
Mbond(2,1)=Xo2;
Mbond(2,2)=Pbo+DPbo2;
Mbond(2,3)=qbo;
Mbond(2,4)=Mbond(1,4)+Mbond(2,2);
Mbond(2,5)=Mbond(1,5)+Mbond(2,3);
Mbond(2,6)=DUbvolo+DUbvolo2;
Mbond(2,7)=DUbso;
Mbond(2,8)=Mbond(2,6)+Mbond(2,7);
Mbond(2,9)=Mbond(1,9)+Mbond(2,8);
Mbond(2,10)=Lo2;
Mbond(2,11)=Kbo2;
Mbond(2,12)=Gbo2;
Mmvol(2,1)=VMo;
Mmvol(2,2)=Vmo;
Mmvol(2,3:4)=0;
Mdefor(2,1)=DEMvo;
Mdefor(2,2)=DEmvo;
Mdefor(2,3)=DEextvo;
Mdefor(2,4)=DEMeso;
Mdefor(2,5)=DEMvo;
Mdefor(2,6)=DEmvo;
Mdefor(2,7)=DEextvo;
Mdefor(2,8)=0;
Mdefor(2,9)=0;
Mdefor(2,10)=DEtotso;
Mdefor(2,11)=DEbvo+DEbvo2;
Mdefor(2,12)=DEbso;
envol(2,1)=M*pqpo(2,1)+M*ks*s;

%%
%Aplicación del incremento que alcance la cedencia en el material%
%Paso1: Determinación de la compresibilidad para una succión s%
ls=lo*((1-r)*exp(-beta*s)+r);
%Paso2: Determinación del esfuerzo de preconsolidación para la succión s%
PMOB=PMc*(PM0/PMc)^((lo-kappaM)/(ls-kappaM));
%Paso2a: verificación del esfuerzo inicial%

```

```

PMOini=PMo+qMo^2/(M^2*(PMo+ks*s));
ev=0;
if PMOini>=PMOB;%n%
    disp('!Error; disminuya el valor del esfuerzo vertical inicial');
    ev=1;
else
    %Paso3: Cálculo de coordenadas (pB,qB) del primer punto de cedencia B%
    %Cálculo de las coordenadas de qB y pB%
    gamma=1+RG2*C2*Cb*C2;
    etha=1+RKb2*C2*Cb*C2*RKe2;
    PTEE=(3*(1-Ko))/(1+2*Ko);%n%
    PTM=(PTEE*etha)/gamma;%n%
    a=1+M^2/(PTM^2);
    b=M^2*(-PMOB+ks*s)/PTM;
    c=M^2*(ks*s)*(-PMOB);
    qMB=(-b+((b^2)-4*a*c)^0.5)/(2*a);
    pMB=PMo+(qMB-qMo)/PTM;%n%
    %Cálculo de los esfuerzos externos%
    qextB=qMB+RG2*(C2^2)*Cb*qMB;
    pextB=Po+(qextB-qo)/PTEE;%n%
    %Cálculo de volúmenes para el segundo incremento%
    if pMB<PMc;
        VMB=VMo;
        VmB=Vmo;
    elseif PMo<PMc && pMB>PMc;
        VMB=VMo-kappaM*log(pMB/PMc);
        VmB=Vmo-kappam*log(pMB/PMc);
    else
        VMB=VMo-kappaM*log(pMB/PMo);
        VmB=Vmo-kappam*log(pMB/PMo);
    end;
    %Paso5: Deformación cortante elástica primer incremento, aplica solamente%
    %macroestructura%
    DEMesB=(qMB-qMo)/(3*GM);%n%
    %Paso6: Verificación de la condición de falla%
    qMFB=(M*pMB+M*ks*s);
    vc=3;
    %Paso7: Vol para el esfuerzo de preconsolidación en B%
    if PMOB<PMc;
        VMoB=VMoB;
        VmoB=VmoB;
    else
        if PMo>PMc
            VMoB=VMo-kappaM*log(PMOB/PMo);
            VmoB=Vmo-kappam*log(PMOB/PMo);
        else
            VMoB=VMo-kappaM*log(PMOB/PMc);
            VmoB=Vmo-kappam*log(PMOB/PMc);
        end
    end;
    end;
    DEMvB=(VMo-VMoB)/VMo;

```

```

DEmvB=(Vmo-VmoB)/Vmo;
DEextvB=DEmvB+DEmvB;
%Paso8: Deformaciones, esfuerzos y energía en el cemento%
DEbvB=DEmvB*C2;
DEbsB=DEmesB*C2;
DPbB=Kbo*DEbvB;
DqbB=Gbo*DEbsB;
pbB=Pbo+DPbB;
qB=qbo+DqbB;%n%
UbpB=pbB*DEbvB;
UqbB=qB*DEbsB;
UbtB=Ubo+UbpB+UqbB;
%Paso9: variable de daño, modulos y relaciones modulares%
if Cb==0;
    LB=0;
elseif log(UbtB/ro)/r1<0;
    LB=0;
else
    LB=log(UbtB/ro)/r1;
end
XB=Xo*exp(-LB/2);
KbB=Kbini*exp(-LB);
GbB=Gbini*exp(-LB);
C3=XB/(1+Cb*XB);
if pMB<PMc;
    KMB=(PMc*VMB)/kappaM;
    KmB=((PMc+s)*VmB)/kappam;
    KMextB=(KMB*KmB)/(KMB+KmB);
else
    KMB=(pMB*VMB)/kappaM;
    KmB=((pMB+s)*VmB)/kappam;
    KMextB=(KMB*KmB)/(KMB+KmB);
end;
RG3=GbB/(3*GM);
RKb3=KbB/KMo;
RKe3=KMextB/KMB;
pqpo(3,1)=pMB;
pqpo(3,2)=qMB;
pqpo(3,3)=PMoB;
envol(3,1)=M*pqpo(3,1)+M*ks*s;
pqext(3,1)=pextB;
pqext(3,2)=qextB;
KMm(3,1)=KMB;
KMm(3,2)=KmB;
KMm(3,3)=KMextB;
const(3,1)=C3;
const(3,2)=RG3;
const(3,3)=RKb3;
const(3,4)=RKe3;
PM(3,1)=gamma;
PM(3,2)=etha;

```

```

PM(3,3)=PTM;
PM(3,4)=qMB-qMo;%n%
Mbond(3,1)=XB;
Mbond(3,2)=DPbB;
Mbond(3,3)=DqbB;
Mbond(3,4)=Mbond(2,4)+Mbond(3,2);
Mbond(3,5)=Mbond(2,5)+Mbond(3,3);
Mbond(3,6)=UbpB;
Mbond(3,7)=UbpqB;
Mbond(3,8)=UbtB;
Mbond(3,9)=UbtB;
Mbond(3,10)=LB;
Mbond(3,11)=KbB;
Mbond(3,12)=GbB;
Mmvol(3,1)=VMB;
Mmvol(3,2)=VmB;
Mmvol(3,3)=VMoB;
Mmvol(3,4)=VmoB;
Mdefor(3,1)=DEmvB;
Mdefor(3,2)=DEmvB;
Mdefor(3,3)=DEextvB;
Mdefor(3,4)=DEMesB;
Mdefor(3,5)=Mdefor(3,1);
Mdefor(3,6)=Mdefor(3,2);
Mdefor(3,7)=Mdefor(3,5)+Mdefor(3,6);
Mdefor(3,8)=0;
Mdefor(3,9)=0;
Mdefor(3,10)=Mdefor(3,4)+Mdefor(3,9);
Mdefor(3,11)=DEbvB;
Mdefor(3,12)=DEbsB;
pqext(3,3)=pqext(2,3)+Mdefor(3,7);
pqext(3,4)=pqext(2,4)+Mdefor(3,10);
%Factor de corrección de Ohmaki%
alf=(M*(M-9)*(M-3))/(9*(6-M)*(1-kappaM/lo));
end

%%
%Aplicación del i-ésimo incremento de esfuerzo%
Pf=sigmavf*(1+2*Ko)/3;%nuevo%
qf=sigmavf*(1-Ko);%nuevo%
if qextB>qf;%n%
    disp('!Error;: Aumente el valor del esfuerzo vertical final debe ser mayor a PM0B');
    ev=2;
end
if ev==1 || ev==2;%n%
    disp('!{Reinicie la aplicación cambiando los valores mencionados}');
else
    %Aplicación de el i-esimo incremento de esfuerzos%
    vc=vc+1;
    fprintf('\n Ingrese el número de incrementos a considerar \n\n');%n%
    n=input('Ingrese n= ');

```



```

Dqext=(qf-qextB)/n;%n%
pqext(vc,2)=pqext(vc-1,2)+Dqext;
pqext(vc,1)=pqext(vc-1,1)+(Dqext/PTEE);%n%
%Pasol1: Cálculo de los i-ésimos esfuerzos en la matriz y en el cemento%
PM(vc,4)=(pqext(vc,2)-pqpo(vc-1,2)-Mbond(vc-1,5)*Cb*const(vc-1,1))/(1+const(vc-1,2)
*const(vc-1,1)*const(vc-1,1)*Cb);
PM(vc,1)=(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
PM(vc,2)=1+const(vc-1,3)*const(vc-1,1)*Cb*const(vc-1,1)*const(vc-1,4);
PM(vc,3)=(PTEE*PM(vc,2))/PM(vc,1);%n%
pqpo(vc,2)=pqpo(vc-1,2)+PM(vc,4);
pqpo(vc,1)=pqpo(vc-1,1)+(PM(vc,4)/PM(vc,3));
Mbond(vc,3)=const(vc-1,2)*const(vc-1,1)*PM(vc,4);
Mbond(vc,2)=const(vc-1,3)*const(vc-1,1)*(PM(vc,4)/PM(vc,3));
Mbond(vc,5)=Mbond(vc-1,5)+Mbond(vc,3);
Mbond(vc,4)=Mbond(vc-1,4)+Mbond(vc,2);
envol(vc,1)=M*pqpo(vc,1)+M*ks*s;
for i=1:n;%n%
    %Pasol2: Esfuerzo de preconsolidación i-ésimo%
    pqpo(vc,3)=pqpo(vc,1)+pqpo(vc,2)^2/(M^2*(pqpo(vc,1)+ks*s));
    %Pasol3: Volúmenes correspondientes al esfuerzo de%
    %preconsolidación%
    Mmvol(vc,3)=Mmvol(vc-1,3)-ls*log(pqpo(vc,3)/pqpo(vc-1,3));
    Mmvol(vc,4)=Mmvol(vc-1,4)-kappam*log(pqpo(vc,3)/pqpo(vc-1,3));
    %Pasol4: Volúmenes correspondientes al esfuerzo i-ésimo en la%
    %matriz%
    if Mmvol(vc,3)+kappaM*log(pqpo(vc,3)/pqpo(vc,1))>Mmvol(vc-1,1);
        Mmvol(vc,1)=Mmvol(vc-1,1);
    else
        Mmvol(vc,1)=Mmvol(vc,3)+kappaM*log(pqpo(vc,3)/pqpo(vc,1));
    end
    if Mmvol(vc,4)+kappam*log(pqpo(vc,3)/pqpo(vc,1))>Mmvol(vc-1,2);
        Mmvol(vc,2)=Mmvol(vc-1,2);
    else
        Mmvol(vc,2)=Mmvol(vc,4)+kappam*log(pqpo(vc,3)/pqpo(vc,1));
    end
    %Pasol5: Incremento en las deformaciones elásticas%
    Mdefor(vc,1)=(pqpo(vc,1)-pqpo(vc-1,1))/KMm(vc-1,1);
    Mdefor(vc,2)=(pqpo(vc,1)-pqpo(vc-1,1))/KMm(vc-1,2);
    Mdefor(vc,3)=Mdefor(vc,1)+Mdefor(vc,2);
    Mdefor(vc,4)=(pqpo(vc,2)-pqpo(vc-1,2))/(3*GM);
    %Pasol6: Deformaciones totales volumétricas%
    Mdefor(vc,5)=(Mmvol(vc-1,1)-Mmvol(vc,1))/Mmvol(vc-1,1);
    Mdefor(vc,6)=(Mmvol(vc-1,2)-Mmvol(vc,2))/Mmvol(vc-1,2);
    Mdefor(vc,7)=Mdefor(vc,5)+Mdefor(vc,6);
    %si defor externas son ceros utilizar proceso alternativo%
    if Mdefor(vc,7)==0;
        Mdefor(vc,11)=Mbond(vc,2)/Mbond(vc-1,11);
        Mdefor(vc,12)=Mbond(vc,3)/Mbond(vc-1,12);
        Mdefor(vc,10)=Mdefor(vc,12)/const(vc-1,1);
        Mdefor(vc,4)=PM(vc,4)/(3*GM);
        Mdefor(vc,9)=Mdefor(vc,10)-Mdefor(vc,4);
    end
end

```

```

Mdefor(vc,8)=Mdefor(vc,9)/((2*alf*pqpo(vc,2))/(M^2*(2*pqpo(vc,1)+ks*s-pqpo
(vc,3)))));
Mdefor(vc,7)=Mdefor(vc,11)/(const(vc-1,1)*const(vc-1,4));
Mdefor(vc,5)=Mdefor(vc,11)/const(vc-1,1);
Mdefor(vc,6)=Mdefor(vc,7)-Mdefor(vc,5);
Mdefor(vc,1)=Mdefor(vc,5)-Mdefor(vc,8);
Mdefor(vc,2)=Mdefor(vc,6);
Mdefor(vc,3)=Mdefor(vc,1)+Mdefor(vc,2);
Mmvol(vc,1)=Mmvol(vc-1,1)-Mdefor(vc,5)*Mmvol(vc-1,1);
Mmvol(vc,2)=Mmvol(vc-1,2)-Mdefor(vc,6)*Mmvol(vc-1,2);
else
%Paso17: Deformaciones plásticas volumétricas%
if Mdefor(vc,5)-Mdefor(vc,1)<0;
    Mdefor(vc,8)=0;
else
    Mdefor(vc,8)=Mdefor(vc,5)-Mdefor(vc,1);
end
%Paso18: Deformaciones plásticas cortantes%
Mdefor(vc,9)=Mdefor(vc,8)*(2*alf*pqpo(vc,2))/(M^2*(2*pqpo(vc,1)+ks*s-pqpo
(vc,3)))));
%Paso19: Deformación cortante total%
Mdefor(vc,10)=Mdefor(vc,4)+Mdefor(vc,9);
%Paso20: Deformaciones, esfuerzos y energía en el cemento%
Mdefor(vc,11)=Mdefor(vc,5)*const(vc-1,1);
Mdefor(vc,12)=Mdefor(vc,10)*const(vc-1,1);
end
Mbond(vc,6)=Mbond(vc,4)*Mdefor(vc,11);
Mbond(vc,7)=Mbond(vc,5)*Mdefor(vc,12);
Mbond(vc,8)=Mbond(vc,6)+Mbond(vc,7);
Mbond(vc,9)=Mbond(vc-1,9)+Mbond(vc,8);
%Paso21: Variable de daño, módulos y relaciones modulares%
if Cb==0;
    Mbond(vc,10)=0;
elseif log(Mbond(vc,9)/ro)/r1<0;
    Mbond(vc,10)=0;
else
    Mbond(vc,10)=log(Mbond(vc,9)/ro)/r1;
end;
Mbond(vc,1)=Mbond(1,1)*exp(-Mbond(vc,10)/2);
Mbond(vc,11)=Mbond(1,11)*exp(-Mbond(vc,10));
Mbond(vc,12)=Mbond(1,12)*exp(-Mbond(vc,10));
if pqpo(vc,1)>PMc;
    KMm(vc,1)=pqpo(vc,1)*Mmvol(vc,1)/kappaM;
    KMm(vc,2)=(pqpo(vc,1)+s)*Mmvol(vc,2)/kappam;
    KMm(vc,3)=KMm(vc,1)*KMm(vc,2)/(KMm(vc,1)+KMm(vc,2));
else
    KMm(vc,1)=PMc*Mmvol(vc,1)/kappaM;
    KMm(vc,2)=(PMc+s)*Mmvol(vc,2)/kappam;
    KMm(vc,3)=(KMm(vc,1)*KMm(vc,2))/(KMm(vc,1)+KMm(vc,2));
end
const(vc,1)=Mbond(vc,1)/(1+Cb*Mbond(vc,1));

```

```

const(vc,2)=Mbond(vc,12)/(3*GM);
const(vc,3)=Mbond(vc,11)/KMm(vc,1);
const(vc,4)=KMm(vc,3)/KMm(vc,1);
%Paso22: Deformaciones externas%()
pqext(vc,3)=pqext(vc-1,3)+Mdefor(vc,7);
pqext(vc,4)=pqext(vc-1,4)+Mdefor(vc,10);
%Paso23: Se inicia la siguiente iteración%
vc=vc+1;
%Esfuerzos externos%
pqext(vc,2)=pqext(vc-1,2)+Dqext;
pqext(vc,1)=pqext(vc-1,1)+(Dqext/PTEE);
%Esfuerzos en la matriz%
PM(vc,4)=(pqext(vc,2)-(pqpo(vc-1,2)+Mbond(vc-1,5)*const(vc-1,1)*Cb))/(1+const
(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
PM(vc,1)=(1+const(vc-1,2)*const(vc-1,1)*const(vc-1,1)*Cb);
PM(vc,2)=1+const(vc-1,3)*const(vc-1,1)*Cb*const(vc-1,1)*const(vc-1,4);
PM(vc,3)=(3*PM(vc,2))/PM(vc,1);
pqpo(vc,2)=pqpo(vc-1,2)+PM(vc,4);
pqpo(vc,1)=pqpo(vc-1,1)+(PM(vc,4)/PM(vc,3));
Mbond(vc,3)=const(vc-1,2)*const(vc-1,1)*PM(vc,4);
Mbond(vc,2)=const(vc-1,3)*const(vc-1,1)*(PM(vc,4)/PM(vc,3));
Mbond(vc,5)=Mbond(vc-1,5)+Mbond(vc,3);
Mbond(vc,4)=Mbond(vc-1,4)+Mbond(vc,2);
envol(vc,1)=M*pqpo(vc,1)+M*ks*s;
if (pqpo(vc,2)>=envol(vc,1));
    disp('Se alcanzó la condición de falla');
    break;
end;
end;
end;
for i=1:vc-1;
Vvt(i,1)=Mmvol(i,1)+Mmvol(i,2);
end
%Gráficas de los resultados%
%Curva de compresibilidad de la macro-estructura%
subplot(2,1,1);
semilogx(pqext(2:vc-1,1),Mmvol(2:vc-1,1));
title('Compresibilidad de la macro-estructura');
xlabel('Pext (MPa)');
ylabel('vM (Mpa)');
%Compresibilidad VM+Vm%
subplot(2,1,2);
semilogx(pqext(2:vc-1,1),Vvt(2:vc-1,1));
title('Curva de Compresibilidad total vM+vm');
xlabel('Pext (MPa)');
ylabel('vM+vm');

%%

ened=0;

```

```

fprintf('\n Presione 1 si desea aplicar un ciclo de humedecimiento secado \n\n');
ecs=input(': ');
if ecs==1;
    %%CICLOS DE HUMEDECIMIENTO - SECADO%
    %Adquisición de parámetros adicionales%
    fprintf('\n Ingrese la diferencia entre las succiones de que definen las superficies SI y SD \n\n');
    DsB=input('Ingrese sI-sD= ');
    fprintf('\n Ingrese el parámetro fI0 para la función de interacción fI \n\n');
    fI0=input('Ingrese fI0= ');
    fprintf('\n Ingrese el parámetro fI1 para la función de interacción fI \n\n');
    fI1=input('Ingrese fI1= ');
    fprintf('\n Ingrese el parámetro nI para la función de interacción fI \n\n');
    nI=input('Ingrese nI= ');
    fprintf('\n Ingrese el parámetro fD0 para la función de interacción fD \n\n');
    fD0=input('Ingrese fD0= ');
    fprintf('\n Ingrese el parámetro fD1 para la función de interacción fD \n\n');
    fD1=input('Ingrese fD1= ');
    fprintf('\n Ingrese el parámetro nD para la función de interacción fD \n\n');
    nD=input('Ingrese nD= ');
    fprintf('\n Ingrese el valor de la succión máxima aplicada smax \n\n');
    smax=input('Ingrese smax= ');
    fprintf('\n Ingrese el valor de la succión mínima aplicada smin \n\n');
    smin=input('Ingrese smin= ');
    fprintf('\n Ingrese el valor del esfuerzo vertical neto durante los ciclos h-s \n\n');
    svhs=input('Ingrese svhs= ');
    if (smax-smin)<(DsB);
        disp('Debe aumentar el valor de la succión máxima');
        fprintf('\n Ingrese el nuevo valor de la succión máxima aplicada smax \n\n');
        smax=input('Ingrese smax= ');
    end
else
    disp('No se ejecutará el ciclo de humedecimiento secado');
end
%%
if ecs==1;
    %%Aplicación del esfuerzo inicial%%
    %Paso1: Cálculo de esfuerzo equivalente inicial modulos Bulk iniciales y constantes%
    Pohs=svhs*(1+2*Ko)/3;%nuevo%
    qohs=svhs*(1-Ko);%nuevo%
    KMini=(PMc*VMini)/kappaM;
    Kmini=((PMc)*Vmini)/kappam;%n%
    KMextini=(KMini*Kmini)/(KMini+Kmini);
    Clhs=Xini/(1+Cb*Xini);
    RGlhs=Gbini/(3*GM);
    RKblhs=Kbini/KMini;
    RKelhs=KMextini/KMini;
    %Paso2: Cálculo del incremento en la matriz y cemento a partir de esf
    %externos%%n%
    PMohs=Pohs/(1+(RKblhs*RKelhs*Cb*Clhs^2));

```

```

Pbohs=RKblhs*Clhs*PMohs;
qMohs=qohs/(1+(RGlhs*Cb*Clhs^2));
qbohs=RGlhs*Clhs*qMohs;%n%
%Paso3: Determinación de los volúmenes para el esfuerzo inicial PMo%%n%
if PMohs<PMc;
    VMohs=VMini;
    Vmohs=Vmini;
else
    VMohs=VMini-kappaM*log(PMohs/PMc);
    Vmohs=Vmini-kappam*log(PMohs/PMc);
end;
%Paso4: Determinación de las deformaciones por el esfuerzo inicial%n%
DEMvohs=(VMini-VMohs)/VMini;
DEmvohs=(Vmini-Vmohs)/Vmini;
DEMesohs=qMohs/(3*GM);
%Paso5: Determinación de la deformación externa por el esfuerzo inicial%
DEextvohs=DEmvohs+DEmvohs;
%Paso6: Determinación de la deformación volumétrica del cemento por el%
%esfuerzo inicial%
if DEMvohs<=0;
    DEbvohs=Pbohs/Kbini;
    DEbsohs=qbohs/Gbini;%n%
    DETotsohs=DEbsohs/Clhs;%n%
    DEMesohs=DETotsohs;%n%
    DEextvosh=DEbvohs/(Clhs*RKelhs);
    DEMvohs=DEbvohs/Xini+Cb*DEbvohs;
    DEMvohs=DEextvosh-DEMvohs;
    VMohs=VMini-DEMvohs*VMini;
    Vmohs=Vmini-DEMvohs*Vmini;
else
    DEbvohs=Xini*DEMvohs/(1+Xini*Cb);
    DEbsohs=Clhs*DEMesohs;%n%
    DETotsohs=DEMesohs;%n%
end;
%Paso7: Determinación del incremento en el esfuerzo equivalente P en el%
%cemento y el cálculo de la energía%
DPbohs=Kbini*DEbvohs;
Pbini=0;
Pbohs=Pbini+DPbohs;
Dqbohs=Gbini*DEbsohs;%n%
qbini=0;%n%
qbohs=qbini+Dqbohs;%n%
DÜbvolohs=Pbohs*DEbvohs/2;%n%
DÜbsohs=qbohs*DEbsohs/2;%n%
DÜbohs=DÜbvolohs+DÜbsohs;
Übini=0;
Übohs=Übini+DÜbohs;
%Paso8: Determinación de la variable de daño L y los módulos del cemento%
if Cb==0
    Lohs=0;
elseif log(Übohs/ro)/r1<0;

```

```

Lohs=0;
else
    Lohs=log(Ubohs/ro)/r1;
end
Xohs=Xini*exp(-Lohs/2);
Kbohs=Kbini*exp(-Lohs);
Gbohs=Gbini*exp(-Lohs);
C2hs=Xohs/(1+Cb*Xohs);
if PMohs<PMc;
    KMohs=(PMc*VMohs)/kappaM;
    Kmohs=((PMc)*Vmohs)/kappam;
    KMextohs=(KMohs*Kmohs)/(KMohs+Kmohs);
else
    KMohs=(PMohs*VMohs)/kappaM;
    Kmohs=((PMohs)*Vmohs)/kappam;
    KMextohs=(KMohs*Kmohs)/(KMohs+Kmohs);
end;
RG2hs=Gbohs/(3*GM);
RKb2hs=Kbohs/KMohs;
RKe2hs=KMextohs/KMohs;
%Paso9: Determinación de las deformaciones por la succión inicial%
Ksini=((Patm)*VMohs)/kappas;
DEMvohs=DEMvohs+(smin/Ksini);
DEmvohs=DEmvohs+(smin/Kmohs);
DEextvohs=DEextvohs+(smin/Ksini)+(smin/Kmohs);
VMohs=VMohs-kappas*log((smin+Patm)/Patm);
Vmohs=Vmohs-kappam*log((smin+PMohs)/PMohs);
%Paso10: Determinación de las deformaciones o esfuerzos en el cemento%
DEbvohs2=C2hs*(smin/Ksini+smin/Kmohs);
DPbohs2=Kbohs*DEbvohs2;
%Paso11: Determinación de la nueva energía de deformación%
DÜbvolohs2=DPbohs2*DEbvohs2;
Ubohs=Ubohs+DÜbvolohs2;
if Cb==0
    Lohs2=0;
elseif log(Ubohs/ro)/r1<0;
    Lohs2=0;
else
    Lohs2=log(Ubohs/ro)/r1;
end
Xohs2=Xini*exp(-Lohs2/2);
Kbohs2=Kbini*exp(-Lohs2);
Gbohs2=Gbini*exp(-Lohs2);
C2hs2=Xohs2/(1+Cb*Xohs2);
if PMohs<PMc;
    KMohs=(PMc*VMohs)/kappaM;
    Kmohs=((PMc+smin)*Vmohs)/kappam;
    KMextohs=(KMohs*Kmohs)/(KMohs+Kmohs);
    Ksohs=((Patm+smin)*VMohs)/kappas;
else
    KMohs=(PMohs*VMohs)/kappaM;

```

```

    Kmohs=( (PMohs+smin)*Vmohs)/kappam;
    KMextohs=(KMohs*Kmohs)/(KMohs+Kmohs);
    Ksohs=((Patm+smin)*VMohs)/kappas;
end;
RG2hs2=Gbohs2/(3*GM);
RKb2hs2=Kbohs2/KMohs;
RKe2hs2=KMextohs/KMohs;
%Asignaciones a las matrices%
pqexths(1,1:5)=0;
KMMhs(1,1)=KMini;
KMMhs(1,2)=Kmini;
KMMhs(1,3)=KMextini;
KMMhs(1,4)=Ksini;
consths(1,1)=Clhs;
consths(1,2)=RGlhs;
consths(1,3)=RKblhs;
consths(1,4)=RKe1hs;
PMhs(1,1:4)=0;
pppohs(1,1:3)=0;
Mbonds(1,1)=Xini;
Mbonds(1,2:9)=0;
Mbonds(1,10)=0;
Mbonds(1,11)=Kbini;
Mbonds(1,12)=Gbini;
Mmvolhs(1,1)=VMini;
Mmvolhs(1,2)=Vmini;
Mmvolhs(1,3:4)=0;
Mdeforhs(1,1:12)=0;
envolhs(1,1)=M*ks*smin;
pqexths(2,1)=Pohs;
pqexths(2,2)=qohs;
pqexths(2,3)=DEextvohs;
pqexths(2,4)=DEMesohs;
pqexths(2,5)=smin;
KMMhs(2,1)=KMohs;
KMMhs(2,2)=Kmohs;
KMMhs(2,3)=KMextohs;
KMMhs(2,4)=Ksohs;
consths(2,1)=C2hs2;
consths(2,2)=RG2hs2;
consths(2,3)=RKb2hs2;
consths(2,4)=RKe2hs2;
PMhs(2,1:4)=0;
pppohs(2,1)=PMohs;
pppohs(2,2)=qMohs;
pppohs(2,3)=0;
Mbonds(2,1)=Xohs2;
Mbonds(2,2)=Pbohs+DPbohs2;
Mbonds(2,3)=qbohs;
Mbonds(2,4)=Mbonds(1,4)+Mbonds(2,2);
Mbonds(2,5)=Mbonds(1,5)+Mbonds(2,3);

```

```

Mbondhs(2,6)=DUbvolohs+DUbvolohs2;
Mbondhs(2,7)=DUbsohs;
Mbondhs(2,8)=Mbondhs(2,6)+Mbondhs(2,7);
Mbondhs(2,9)=Mbondhs(1,9)+Mbondhs(2,8);
Mbondhs(2,10)=Lohs2;
Mbondhs(2,11)=Kbohs2;
Mbondhs(2,12)=Gbohs2;
Mmvolhs(2,1)=VMohs;
Mmvolhs(2,2)=Vmohs;
Mmvolhs(2,3:4)=0;
Mdeforhs(2,1)=DEMvohs;
Mdeforhs(2,2)=DEmvohs;
Mdeforhs(2,3)=DEextvohs;
Mdeforhs(2,4)=DEMsohs;
Mdeforhs(2,5)=DEMvohs;
Mdeforhs(2,6)=DEmvohs;
Mdeforhs(2,7)=DEextvohs;
Mdeforhs(2,8)=0;
Mdeforhs(2,9)=0;
Mdeforhs(2,10)=DEtotsohs;
Mdeforhs(2,11)=DEbvohs+DEbvohs2;
Mdeforhs(2,12)=DEbsohs;
envolhs(2,1)=M*pppohs(2,1)+M*ks*smin;
end

%%
if ecs==1;
    %Aplicación del incremento en la succión que produce la cedencia en el material%
    %Determinación de la succión que determina la cedencia del material%
    sB=smin+(DsB/2);
    pqexths(3,5)=sB;
    j=3;
    fprintf('\n Ingrese el número de incrementos en la succión \n\n');
    nis=input('Ingrese nis= ');
    %Valor del incremento en la succión%
    Dsc=(smax-sB)/nis;
    %Determinación de la compresibilidad y el esfuerzo PMo(s)%
    lsmin=lo*((1-r)*exp(-beta*smin)+r);
    pppohs(3,3)=PMc*(PMo/PMc)^((lo-kappaM)/(lsmin-kappaM));
    %Valor de la función de interacción fi%
    fi(j,1)=fI0+fI1*((PMohs/pppohs(j,3))^nI);
    cds=0;
    %Volumenes correspondientes a la succión de cedencia%
    Mmvolhs(j,1)=Mmvolhs(j-1,1)-kappas*log(sB/smin);
    Mmvolhs(j,2)=Mmvolhs(j-1,2)-kappas*log(sB/smin);
    Mmvolhs(j,3)=0;
    Mmvolhs(j,4)=0;
    %Deformaciones correspondientes a la succión de cedencia%
    Mdeforhs(j,1)=(Mmvolhs(j-1,1)-Mmvolhs(j,1))/Mmvolhs(j-1,1);
    Mdeforhs(j,2)=(Mmvolhs(j-1,2)-Mmvolhs(j,2))/Mmvolhs(j-1,2);
    Mdeforhs(j,3)=Mdeforhs(j,1)+Mdeforhs(j,2);

```



```

Mdeforhs(j,4)=0;
Mdeforhs(j,5)=Mdeforhs(j,1);
Mdeforhs(j,6)=Mdeforhs(j,2);
Mdeforhs(j,7)=Mdeforhs(j,5)+Mdeforhs(j,6);
Mdeforhs(j,8)=Mdeforhs(j,5)-Mdeforhs(j,1);
Mdeforhs(j,9)=0;
Mdeforhs(j,10)=0;
Mdeforhs(j,11)=consths(j-1,1)*Mdeforhs(j,5);
Mdeforhs(j,12)=consths(j-1,1)*Mdeforhs(j,10);
%Evolución de la varibale de daño%
Mbondhs(j,2)=Mbondhs(j-1,11)*Mdeforhs(j,11);
Mbondhs(j,3)=Mbondhs(j-1,12)*Mdeforhs(j,12);
Mbondhs(j,4)=Mbondhs(j-1,4)+Mbondhs(j,2);
Mbondhs(j,5)=Mbondhs(j-1,5)+Mbondhs(j,3);
Mbondhs(j,6)=Mbondhs(j,4)*Mdeforhs(j,11);
Mbondhs(j,7)=Mbondhs(j,5)*Mdeforhs(j,12);
Mbondhs(j,8)=Mbondhs(j,7)+Mbondhs(j,6);
Mbondhs(j,9)=Mbondhs(j-1,9)+Mbondhs(j,8);
if Cb==0;
    Mbondhs(j,10)=0;
elseif log(Mbondhs(j,9)/ro)/r1<Mbondhs(j-1,10);
    Mbondhs(j,10)=Mbondhs(j-1,10);
else
    Mbondhs(j,10)=log(Mbondhs(j,9)/ro)/r1;
end;
Mbondhs(j,1)=Mbondhs(1,1)*exp(-Mbondhs(j,10)/2);
Mbondhs(j,11)=Mbondhs(1,11)*exp(-Mbondhs(j,10));
Mbondhs(j,12)=Mbondhs(1,12)*exp(-Mbondhs(j,10));
pqexths(j,3)=pqexths(j-1,3)+Mdeforhs(j,7);
consths(j,1)=Mbondhs(j,1)/(1+Cb*Mbondhs(j,1));
t=1+j;
%aplicación de los incrementos en la succión hasta llegar a smax%
for j=t:(2*nis+5);
    if pqexths(j-1,5)>smax;
        disp('Se alcanzó la succión máxima');
        cds=1;
    end
    if cds==0;
        pqexths(j,5)=pqexths(j-1,5)+Dsc;
    else
        pqexths(j,5)=pqexths(j-1,5)-Dsc;
    end;
    lsj=lo*((1-r)*exp(-beta*pqexths(j,5))+r);
    pqpohs(j,3)=PMc*(PM0/PMc)^((lo-kappaM)/(lsj-kappaM));
    fi(j,1)=fI0+fI1*((PMohs/pqpohs(j,3))^nI);
    fd(j,1)=fD0+fD1*((1-PMohs/pqpohs(j,3))^nD);
    Mmvolhs(j,1)=Mmvolhs(j-1,1)-kappas*log(pqexths(j,5)/pqexths(j-1,5));
    Mmvolhs(j,2)=Mmvolhs(j-1,2)-kappas*log(pqexths(j,5)/pqexths(j-1,5));
    Mmvolhs(j,3)=0;
    Mmvolhs(j,4)=0;
    %Cálculo de deformaciones%

```

```

Mdeforhs(j,1)=(Mmvolhs(j-1,1)-Mmvolhs(j,1))/Mmvolhs(j-1,1);
Mdeforhs(j,2)=(Mmvolhs(j-1,2)-Mmvolhs(j,2))/Mmvolhs(j-1,2);
Mdeforhs(j,3)=Mdeforhs(j,1)+Mdeforhs(j,2);
Mdeforhs(j,4)=0;
if cds==0;
    Mdeforhs(j,5)=Mdeforhs(j,1)+fi(j,1)*Mdeforhs(j,2);
else
    Mdeforhs(j,5)=Mdeforhs(j,1)+fd(j,1)*Mdeforhs(j,2);
end
Mdeforhs(j,6)=Mdeforhs(j,2);
Mdeforhs(j,7)=Mdeforhs(j,5)+Mdeforhs(j,6);
Mdeforhs(j,8)=Mdeforhs(j,5)-Mdeforhs(j,1);
Mdeforhs(j,9)=0;
Mdeforhs(j,10)=0;
Mdeforhs(j,11)=consths(j-1,1)*Mdeforhs(j,5);
Mdeforhs(j,12)=consths(j-1,1)*Mdeforhs(j,10);
%se corrije el volumen macro por deformaciones plásticas%
Mmvolhs(j,1)=Mmvolhs(j-1,1)-Mdeforhs(j,5)*Mmvolhs(j-1,1);
%Evolución de la varibale de daño%
Mbondhs(j,2)=Mbondhs(j-1,11)*Mdeforhs(j,11);
Mbondhs(j,3)=Mbondhs(j-1,12)*Mdeforhs(j,12);
Mbondhs(j,4)=Mbondhs(j-1,4)+Mbondhs(j,2);
Mbondhs(j,5)=Mbondhs(j-1,5)+Mbondhs(j,3);
Mbondhs(j,6)=Mbondhs(j,4)*Mdeforhs(j,11);
Mbondhs(j,7)=Mbondhs(j,5)*Mdeforhs(j,12);
Mbondhs(j,8)=Mbondhs(j,7)+Mbondhs(j,6);
Mbondhs(j,9)=Mbondhs(j-1,9)+Mbondhs(j,8);
if Cb==0;
    Mbondhs(j,10)=0;
elseif log(Mbondhs(j,9)/ro)/r1<Mbondhs(j-1,10);
    Mbondhs(j,10)=Mbondhs(j-1,10);
else
    Mbondhs(j,10)=log(Mbondhs(j,9)/ro)/r1;
end;
Mbondhs(j,1)=Mbondhs(1,1)*exp(-Mbondhs(j,10)/2);
Mbondhs(j,11)=Mbondhs(1,11)*exp(-Mbondhs(j,10));
Mbondhs(j,12)=Mbondhs(1,12)*exp(-Mbondhs(j,10));
consths(j,1)=Mbondhs(j,1)/(1+Cb*Mbondhs(j,1));
pqexths(j,3)=pqexths(j-1,3)+Mdeforhs(j,7);
end
%Se hace la descarga hasta el estado de esfuerzos inicial%
j=j+1;
pqexths(j,5)=smin;
lsj=lo*((1-r)*exp(-beta*pqexths(j,5))+r);
pppohs(j,3)=PMc*(PM0/PMc)^((lo-kappaM)/(lsj-kappaM));
fd(j,1)=fD0+fD1*((1-PMohs/pppohs(j,3))^nI);
Mmvolhs(j,1)=Mmvolhs(j-1,1)-kappas*log(pqexths(j,5)/pqexths(j-1,5));
Mmvolhs(j,2)=Mmvolhs(j-1,2)-kappas*log(pqexths(j,5)/pqexths(j-1,5));
Mmvolhs(j,3)=0;
Mmvolhs(j,4)=0;
%Cálculo de deformaciones%

```

```

Mdeforhs(j,1)=(Mmvolhs(j-1,1)-Mmvolhs(j,1))/Mmvolhs(j-1,1);
Mdeforhs(j,2)=(Mmvolhs(j-1,2)-Mmvolhs(j,2))/Mmvolhs(j-1,2);
Mdeforhs(j,3)=Mdeforhs(j,1)+Mdeforhs(j,2);
Mdeforhs(j,4)=0;
Mdeforhs(j,5)=Mdeforhs(j,1);
Mdeforhs(j,6)=Mdeforhs(j,2);
Mdeforhs(j,7)=Mdeforhs(j,5)+Mdeforhs(j,6);
Mdeforhs(j,8)=Mdeforhs(j,5)-Mdeforhs(j,1);
Mdeforhs(j,9)=0;
Mdeforhs(j,10)=0;
Mdeforhs(j,11)=consths(j-1,1)*Mdeforhs(j,5);
Mdeforhs(j,12)=consths(j-1,1)*Mdeforhs(j,10);
%Evolución de la varibale de daño%
Mbondhs(j,2)=Mbondhs(j-1,11)*Mdeforhs(j,11);
Mbondhs(j,3)=Mbondhs(j-1,12)*Mdeforhs(j,12);
Mbondhs(j,4)=Mbondhs(j-1,4)+Mbondhs(j,2);
Mbondhs(j,5)=Mbondhs(j-1,5)+Mbondhs(j,3);
Mbondhs(j,6)=Mbondhs(j,4)*Mdeforhs(j,11);
Mbondhs(j,7)=Mbondhs(j,5)*Mdeforhs(j,12);
Mbondhs(j,8)=Mbondhs(j,7)+Mbondhs(j,6);
Mbondhs(j,9)=Mbondhs(j-1,9)+Mbondhs(j,8);
if Cb==0;
    Mbondhs(j,10)=0;
elseif log(Mbondhs(j,9)/ro)/r1<Mbondhs(j-1,10);
    Mbondhs(j,10)=Mbondhs(j-1,10);
else
    Mbondhs(j,10)=log(Mbondhs(j,9)/ro)/r1;
end;
Mbondhs(j,1)=Mbondhs(1,1)*exp(-Mbondhs(j,10)/2);
Mbondhs(j,11)=Mbondhs(1,11)*exp(-Mbondhs(j,10));
Mbondhs(j,12)=Mbondhs(1,12)*exp(-Mbondhs(j,10));
pqexths(j,3)=pqexths(j-1,3)+Mdeforhs(j,7);
consths(j,1)=Mbondhs(j,1)/(1+Cb*Mbondhs(j,1));
end
%%
if ecs==1;
    fprintf('\n Presione 1 si desea realizar un nuevo ensayo edometrico considerando los
\n\n');
    fprintf('\n parámetros X, Kb y Gb obtenidos al final del ciclo humedecimiento secado
\n\n');
    fprintf('\n presione cualquier tecla si desea salir \n\n');
    ened=input(': ');
end;
if ened==1;
    %NUEVO ENSAYO DE CONSOLIDACIÓN MATERIAL DEGRADADO%
    %Características iniciales del material%
    k=1;
    fprintf('\n Se asume que se conservan las características iniciales de la
matriz\n\n');
    fprintf('en aras de valorar el efecto de los ciclos humedecimiento-secado \n\n');
    Mmvol2(k,1:4)=Mmvol(1,1:4);

```

```

pqext2(k,1:5)=0;
pqpo2(k,1:3)=0;
Mdefor2(k,1:12)=0;
envol2(k,1)=envol(1,1);
KMm2(k,1:3)=KMm(1,1:3);
Mbond2(k,1:12)=0;
fprintf('\n Se toman de la etapa de ciclos h-s los parámetros: X, Kb y Gb\n\n');
Mbond2(k,1)=Mbondhs(j,1);
Mbond2(k,10)=Mbondhs(j,10);
Mbond2(k,11)=Mbondhs(j,11);
Mbond2(k,12)=Mbondhs(j,12);
const2(k,1)=Mbond2(k,1)/(1+Cb*Mbond2(k,1));
const2(k,2)=Mbond2(k,12)/(3*GM);
const2(k,3)=Mbond2(k,11)/KMm2(k,1);
const2(k,4)=KMm2(k,3)/KMm2(k,1);
PM2(k,1:4)=0;
%Aplicación de los esfuerzos equivalentes iniciales%
k=k+1;
pqext2(k,1)=sigmavo*(1+2*Ko)/3;
pqext2(k,2)=sigmavo*(1-Ko);
pqpo2(k,1)=pqext2(k,1)/(1+const2(k-1,3)*const2(k-1,4)*Cb*const2(k-1,1)^2);
pqpo2(k,2)=pqext2(k,2)/(1+const2(k-1,2)*Cb*const2(k-1,1)^2);
Mbond2(k,2)=const2(k-1,3)*const2(k-1,1)*pqpo2(k,1);
Mbond2(k,3)=const2(k-1,2)*const2(k-1,1)*pqpo2(k,2);
%determinación de los volúmenes correspondientes a los esfuerzos iniciales%
if pqpo2(k,1)<PMc;
    Mmvol2(k,1)=Mmvol2(k-1,1);
    Mmvol2(k,2)=Mmvol2(k-1,2);
else
    Mmvol2(k,1)=Mmvol2(k-1,1)-kappaM*log(pqpo2(k,1)/PMc);
    Mmvol2(k,2)=Mmvol2(k-1,2)-kappaM*log(pqpo2(k,1)/PMc);
end;
%Determinación de las deformaciones producidas por el esfuerzo inicial%
Mdefor2(k,1)=(Mmvol2(k-1,1)-Mmvol2(k,1))/Mmvol2(k-1,1);
Mdefor2(k,2)=(Mmvol2(k-1,2)-Mmvol2(k,2))/Mmvol2(k-1,2);
Mdefor2(k,3)=Mdefor2(k,1)+Mdefor2(k,2);
Mdefor2(k,4)=pqpo2(k,2)/(3*GM);
Mdefor2(k,5)=Mdefor2(k,1);
Mdefor2(k,6)=Mdefor2(k,2);
Mdefor2(k,7)=Mdefor2(k,3);
if Mdefor2(k,7)==0;
    Mdefor2(k,11)=Mbond2(k,2)/Mbond2(k-1,11);
    Mdefor2(k,12)=Mbond2(k,3)/Mbond2(k-1,12);
    Mdefor2(k,10)=Mdefor2(k,12)/const2(k-1,1);
    Mdefor2(k,4)=Mdefor2(k,10);
    Mdefor2(k,8)=0;
    Mdefor2(k,9)=0;
    Mdefor2(k,7)=Mdefor2(k,11)/(const2(k-1,1)*const2(k-1,4));
    Mdefor2(k,5)=Mdefor2(k,11)/const2(k-1,1)+Cb*Mdefor2(k,11);
    Mdefor2(k,6)=Mdefor2(k,7)-Mdefor2(k,5);
    Mdefor2(k,1)=Mdefor2(k,5)-Mdefor2(k,8);

```

```

Mdefor2(k,2)=Mdefor2(k,6);
Mdefor2(k,3)=Mdefor2(k,1)+Mdefor2(k,2);
Mmvol2(k,1)=Mmvol2(k-1,1)-Mdefor2(k,5)*Mmvol2(k-1,1);
Mmvol2(k,2)=Mmvol2(k-1,2)-Mdefor2(k,6)*Mmvol2(k-1,2);
else
Mdefor2(k,8)=0;
Mdefor2(k,9)=0;
Mdefor2(k,10)=Mdefor2(k,4);
Mdefor2(k,11)=Mdefor2(k,5)*const2(k-1,1);
Mdefor2(k,12)=Mdefor2(k,10)*const2(k-1,1);
end
Mbond2(k,4)=Mbond2(k-1,4)+Mbond2(k,2);
Mbond2(k,5)=Mbond2(k-1,5)+Mbond2(k,3);
Mbond2(k,6)=Mbond2(k,4)*Mdefor2(k,11)/2;
Mbond2(k,7)=Mbond2(k,5)*Mdefor2(k,12)/2;
Mbond2(k,8)=Mbond2(k,6)+Mbond2(k,7);
Mbond2(k,9)=Mbond2(k-1,9)+Mbond2(k,8);
if Cb==0;
Mbond2(k,10)=0;
elseif log(Mbond2(k,9)/ro)/r1<Mbond2(k-1,10);
Mbond2(k,10)=Mbond2(k-1,10);
else
Mbond2(k,10)=log(Mbond2(k,9)/ro)/r1;
end
Mbond2(k,1)=Mbond2(1,1)*exp(-Mbond2(k,10)/2);
Mbond2(k,11)=Mbond2(1,11)*exp(-Mbond2(k,10));
Mbond2(k,12)=Mbond2(1,12)*exp(-Mbond2(k,10));
if pppo2(k,1)>PMc;
KMm2(k,1)=pppo2(k,1)*Mmvol2(k,1)/kappaM;
KMm2(k,2)=(pppo2(k,1))*Mmvol2(k,2)/kappam;
KMm2(k,3)=KMm2(k,1)*KMm2(k,2)/(KMm2(k,1)+KMm2(k,2));
else
KMm2(k,1)=PMc*Mmvol2(k,1)/kappaM;
KMm2(k,2)=(PMc)*Mmvol2(k,2)/kappam;
KMm2(k,3)=(KMm2(k,1)*KMm2(k,2))/(KMm2(k,1)+KMm2(k,2));
end
const2(k,1)=Mbond2(k,1)/(1+Cb*Mbond2(k,1));
const2(k,2)=Mbond2(k,12)/(3*GM);
const2(k,3)=Mbond2(k,11)/KMm2(k,1);
const2(k,4)=KMm2(k,3)/KMm2(k,1);
%Determinación de las deformaciones por la succión inicial%
Kso2=(Patm*Mmvol2(k,1))/kappas;
Mdefor2(k,1)=Mdefor2(k,1)+(s/Kso);
Mdefor2(k,2)=Mdefor2(k,2)+(s/KMm2(k,2));
Mmvol2(k,1)=Mmvol2(k,1)-kappas*log((s+pppo2(k,1))/pppo2(k,1));
Mmvol2(k,2)=Mmvol2(k,2)-kappas*log((s+pppo2(k,1))/pppo2(k,1));
Mdefor2(k,3)=Mdefor2(k,1)+Mdefor2(k,2);
Mdefor2(k,5)=Mdefor2(k,1);
Mdefor2(k,6)=Mdefor2(k,2);
Mdefor2(k,7)=Mdefor2(k,3);
Mdefor2(k,11)=Mdefor2(k,11)+const2(k,1)*((s/Kso)+s/KMm2(k,2));

```

```

Mbond2(k,2)=Mbond2(k,2)+Mbond2(k,11)*const2(k,1)*((s/Kso)+s/KMm2(k,2));
Mbond2(k,4)=Mbond2(k-1,4)+Mbond2(k,2);
Mbond2(k,6)=Mbond2(k,6)+(Mbond2(k,11)*const2(k,1)*((s/Kso)+s/KMm2(k,2)))*(const2(k,1)*((s/Kso)+s/KMm2(k,2)));
Mbond2(k,8)=Mbond2(k,6)+Mbond2(k,7);
Mbond2(k,9)=Mbond2(k-1,9)+Mbond2(k,8);
if Cb==0;
    Mbond2(k,10)=0;
elseif log(Mbond2(k,9)/ro)/r1<Mbond2(k-1,10);
    Mbond2(k,10)=Mbond2(k-1,10);
else
    Mbond2(k,10)=log(Mbond2(k,9)/ro)/r1;
end
Mbond2(k,1)=Mbond2(1,1)*exp(-Mbond2(k,10)/2);
Mbond2(k,11)=Mbond2(1,11)*exp(-Mbond2(k,10));
Mbond2(k,12)=Mbond2(1,12)*exp(-Mbond2(k,10));
if pppo2(k,1)>PMc;
    KMm2(k,1)=pppo2(k,1)*Mmvol2(k,1)/kappaM;
    KMm2(k,2)=(pppo2(k,1))*Mmvol2(k,2)/kappam;
    KMm2(k,3)=KMm2(k,1)*KMm2(k,2)/(KMm2(k,1)+KMm2(k,2));
else
    KMm2(k,1)=PMc*Mmvol2(k,1)/kappaM;
    KMm2(k,2)=(PMc)*Mmvol2(k,2)/kappam;
    KMm2(k,3)=(KMm2(k,1)*KMm2(k,2))/(KMm2(k,1)+KMm2(k,2));
end
const2(k,1)=Mbond2(k,1)/(1+Cb*Mbond2(k,1));
const2(k,2)=Mbond2(k,12)/(3*GM);
const2(k,3)=Mbond2(k,11)/KMm2(k,1);
const2(k,4)=KMm2(k,3)/KMm2(k,1);
pqext2(k,3)=pqext2(k-1,3)+Mdefor2(k,7);
pqext2(k,4)=pqext2(k-1,4)+Mdefor2(k,10);
envol2(k,1)=M*pppo2(k,1)+M*ks*s;
end
%%
if ened==1;
    %Aplicación del incremento que alcance la cedencia en el material%
    k=k+1;
    %Paso1: Determinación de la compresibilidad para una succión s%
    ls=lo*((1-r)*exp(-beta*s)+r);
    %Paso2: Determinación del esfuerzo de preconsolidación para la succión s%
    pppo2(k,3)=PMc*(PM0/PMc)^((lo-kappaM)/(ls-kappaM));
    %Paso2a: verificación del esfuerzo inicial%
    pppo2(k-1,3)=pppo2(k-1,1)+pppo2(k-1,2)^2/(M^2*(pppo2(k-1,1)+ks*s));
    ev=0;
    if pppo2(k-1,3)>=pppo2(k,3);%n%
        disp('!Error; disminuya el valor del esfuerzo vertical inicial');
        ev2=1;
    else
        %Paso3: Cálculo de coordenadas (pB,qB) del primer punto de cedencia B%
        %Cálculo de las coordenadas de qB y pB%
        PM2(k,1)=1+const2(k-1,2)*const2(k-1,1)*Cb*const2(k-1,1);

```

```

PM2(k,2)=1+const2(k-1,3)*const2(k-1,1)*Cb*const2(k-1,1)*const2(k-1,4);
PTEE=(3*(1-Ko))/(1+2*Ko);%n%
PM2(k,3)=(PTEE*PM2(k,2))/PM2(k,1);%n%
a2=1+M^2/(PM2(k,3)^2);
b2=M^2*(-pqqo2(k,3)+ks*s)/PM2(k,3);
c2=M^2*(ks*s)*(-pqqo2(k,3));
pqqo2(k,2)=(-b2+((b2^2)-4*a2*c2)^0.5)/(2*a2);
pqqo2(k,1)=pqqo2(k-1,1)+(pqqo2(k,2)-pqqo2(k-1,2))/PM2(k,3);%n%
PM2(k,4)=pqqo2(k,2)-pqqo2(k-1,2);
%Cálculo de los esfuerzos externos%
pqext2(k,2)=pqqo2(k,2)+const2(k-1,2)*(const2(k-1,1)^2)*Cb*pqqo2(k,2);
pqext2(k,1)=pqext2(k-1,1)+(pqext2(k,2)-pqext2(k-1,2))/PTEE;%n%
%Cálculo de volúmenes para el segundo incremento%
if pqqo2(k,1)<PMc;
    Mmvol2(k,1)=Mmvol2(k-1,1);
    Mmvol2(k,2)=Mmvol2(k-1,2);
elseif pqqo2(k-1,1)<PMc && pqqo2(k,1)>PMc;
    Mmvol2(k,1)=Mmvol2(k-1,1)-kappaM*log(pqqo2(k,1)/PMc);
    Mmvol2(k,2)=Mmvol2(k-1,2)-kappam*log(pqqo2(k,1)/PMc);
else
    Mmvol2(k,1)=Mmvol2(k-1,1)-kappaM*log(pqqo2(k,1)/pqqo2(k-1,1));
    Mmvol2(k,2)=Mmvol2(k-1,2)-kappam*log(pqqo2(k,1)/pqqo2(k-1,1));
end;
%Paso5: Deformación cortante elástica primer incremento, aplica sólomente%
%macroestructura%
Mdefor2(k,4)=(pqqo2(k,2)-pqqo2(k-1,2))/(3*GM);%n%
%Paso6: Verificación de la condición de falla%
envol2(k,1)=(M*pqqo2(k,1)+M*ks*s);
%Paso7: Vol para el esfuerzo de preconsolidación en B%
if pqqo2(k,1)<PMc;
    Mmvol2(k,3)=Mmvol2(k-2,1);
    Mmvol2(k,4)=Mmvol2(k-2,2);
else
    if pqqo2(k-1,1)>PMc;
        Mmvol2(k,3)=Mmvol2(k-1,1)-kappaM*log(pqqo2(k,3)/pqqo2(k-1,1));
        Mmvol2(k,4)=Mmvol2(k-1,2)-kappam*log(pqqo2(k,3)/pqqo2(k-1,1));
    else
        Mmvol2(k,3)=Mmvol2(k-1,1)-kappaM*log(pqqo2(k,3)/PMc);
        Mmvol2(k,4)=Mmvol2(k-1,2)-kappam*log(pqqo2(k,3)/PMc);
    end
end;
Mdefor2(k,1)=(Mmvol2(k-1,1)-Mmvol2(k,3))/Mmvol2(k-1,1);
Mdefor2(k,2)=(Mmvol2(k-1,2)-Mmvol2(k,4))/Mmvol2(k-1,2);
Mdefor2(k,3)=Mdefor2(k,1)+Mdefor2(k,2);
Mdefor2(k,5)=Mdefor2(k,1);
Mdefor2(k,6)=Mdefor2(k,2);
Mdefor2(k,7)=Mdefor2(k,5)+Mdefor2(k,6);
Mdefor2(k,8)=0;
Mdefor2(k,9)=0;
Mdefor2(k,10)=Mdefor2(k,4);
%Paso8: Deformaciones, esfuerzos y energía en el cemento%

```

```

Mdefor2(k,11)=Mdefor2(k,1)*const2(k-1,1);
Mdefor2(k,12)=Mdefor2(k,10)*const2(k-1,1);
Mbond2(k,2)=Mbond2(k-1,11)*Mdefor2(k,11);
Mbond2(k,3)=Mbond2(k-1,12)*Mdefor2(k,12);
Mbond2(k,4)=Mbond2(k-1,4)+Mbond2(k,2);
Mbond2(k,5)=Mbond2(k-1,5)+Mbond2(k,3);
Mbond2(k,6)=Mbond2(k,4)*Mdefor2(k,11);
Mbond2(k,7)=Mbond2(k,5)*Mdefor2(k,12);
Mbond2(k,8)=Mbond2(k,6)+Mbond2(k,7);
Mbond2(k,9)=Mbond2(k-1,9)+Mbond2(k,8);
%Variable de daño, módulos y relaciones modulares%
if Cb==0;
    Mbond2(k,10)=0;
elseif log(Mbond2(k,9)/ro)/r1<Mbond2(k-1,10);
    Mbond2(k,10)=Mbond2(k-1,10);
else
    Mbond2(k,10)=log(Mbond2(k,9)/ro)/r1;
end
Mbond2(k,1)=Mbond2(1,1)*exp(-Mbond2(k,10)/2);
Mbond2(k,11)=Mbond2(1,11)*exp(-Mbond2(k,10));
Mbond2(k,12)=Mbond2(1,12)*exp(-Mbond2(k,10));
if pqpo2(k,1)>PMc;
    KMm2(k,1)=pqpo2(k,1)*Mmvol2(k,1)/kappaM;
    KMm2(k,2)=(pqpo2(k,1)+s)*Mmvol2(k,2)/kappam;
    KMm2(k,3)=KMm2(k,1)*KMm2(k,2)/(KMm2(k,1)+KMm2(k,2));
else
    KMm2(k,1)=PMc*Mmvol2(k,1)/kappaM;
    KMm2(k,2)=(PMc+s)*Mmvol2(k,2)/kappam;
    KMm2(k,3)=(KMm2(k,1)*KMm2(k,2))/(KMm2(k,1)+KMm2(k,2));
end
const2(k,1)=Mbond2(k,1)/(1+Cb*Mbond2(k,1));
const2(k,2)=Mbond2(k,12)/(3*GM);
const2(k,3)=Mbond2(k,11)/KMm2(k,1);
const2(k,4)=KMm2(k,3)/KMm2(k,1);
%Deformaciones externas%()
pqext2(k,3)=pqext2(k-1,3)+Mdefor2(k,7);
pqext2(k,4)=pqext2(k-1,4)+Mdefor2(k,10);
end
alf=(M*(M-9)*(M-3))/(9*(6-M)*(1-kappaM/lo));
end
%%
if ened==1;
    %Aplicación del i-ésimo incremento de esfuerzo%
    qf2=sigmavf*(1-Ko);%nuevo%
    ev2=0;
    if pqext2(k,2)>qf2;%n%
        disp('!Error;: Aumente el valor del esfuerzo vertical final debe ser mayor a
pqpo2(3,2)');
        ev2=2;
    end
    if ev2==1 || ev2==2;%n%

```



```

disp('{Reinicie la aplicación cambiando los valores mencionados'});
else
    %Aplicación de el i-esimo incremento de esfuerzos%
    k=k+1;
    Dqext2=(qf2-pqext2(k-1,2))/n;%n%
    pqext2(k,2)=pqext2(k-1,2)+Dqext2;
    pqext2(k,1)=pqext2(k-1,1)+(Dqext2/PTEE);%n%
    %Paso11: Cálculo de los i-ésimos esfuerzos en la matriz y en el cemento%
    PM2(k,4)=(pqext2(k,2)-pqpo2(k-1,2)-Mbond2(k-1,5)*Cb*const2(k-1,1))/(1+const2(k-1,2)*const2(k-1,1)*const2(k-1,1)*Cb);
    PM2(k,1)=(1+const2(k-1,2)*const2(k-1,1)*const2(k-1,1)*Cb);
    PM2(k,2)=1+const2(k-1,3)*const2(k-1,1)*Cb*const2(k-1,1)*const2(k-1,4);
    PM2(k,3)=(PTEE*PM2(k,2))/PM2(k,1);%n%
    pqpo2(k,2)=pqpo2(k-1,2)+PM2(k,4);
    pqpo2(k,1)=pqpo2(k-1,1)+(PM2(k,4)/PM2(k,3));
    Mbond2(k,3)=const2(k-1,2)*const2(k-1,1)*PM2(k,4);
    Mbond2(k,2)=const2(k-1,3)*const2(k-1,1)*(PM2(k,4)/PM2(k,3));
    Mbond2(k,5)=Mbond2(k-1,5)+Mbond2(k,3);
    Mbond2(k,4)=Mbond2(k-1,4)+Mbond2(k,2);
    envol2(k,1)=M*pqpo2(k,1)+M*ks*s;
    for i=1:n;%n%
        %Paso12: Esfuerzo de preconsolidación i-ésimo%
        pqpo2(k,3)=pqpo2(k,1)+pqpo2(k,2)^2/(M^2*(pqpo2(k,1)+ks*s));
        %Paso13: Volúmenes correspondientes al esfuerzo de
        %preconsolidación%
        Mmvol2(k,3)=Mmvol2(k-1,3)-ls*log(pqpo2(k,3)/pqpo2(k-1,3));
        Mmvol2(k,4)=Mmvol2(k-1,4)-kappam*log(pqpo2(k,3)/pqpo2(k-1,3));
        %Paso14: Volúmenes correspondientes al esfuerzo i-ésimo en la
        %matriz%
        if Mmvol2(k,3)+kappaM*log(pqpo2(k,3)/pqpo2(k,1))>Mmvol2(k-1,1);
            Mmvol2(k,1)=Mmvol2(k-1,1);
        else
            Mmvol2(k,1)=Mmvol2(k,3)+kappaM*log(pqpo2(k,3)/pqpo2(k,1));
        end
        if Mmvol2(k,4)+kappam*log(pqpo2(k,3)/pqpo2(k,1))>Mmvol2(k-1,2);
            Mmvol2(k,2)=Mmvol2(k-1,2);
        else
            Mmvol2(k,2)=Mmvol2(k,4)+kappam*log(pqpo2(k,3)/pqpo2(k,1));
        end
        %Paso15: Incremento en las deformaciones elásticas%
        Mdefor2(k,1)=(pqpo2(k,1)-pqpo2(k-1,1))/KMm2(k-1,1);
        Mdefor2(k,2)=(pqpo2(k,1)-pqpo2(k-1,1))/KMm2(k-1,2);
        Mdefor2(k,3)=Mdefor2(k,1)+Mdefor2(k,2);
        Mdefor2(k,4)=(pqpo2(k,2)-pqpo2(k-1,2))/(3*GM);
        %Paso16: Deformaciones totales volumétricas%
        Mdefor2(k,5)=(Mmvol2(k-1,1)-Mmvol2(k,1))/Mmvol2(k-1,1);
        Mdefor2(k,6)=(Mmvol2(k-1,2)-Mmvol2(k,2))/Mmvol2(k-1,2);
        Mdefor2(k,7)=Mdefor2(k,5)+Mdefor2(k,6);
        %si defor externas son ceros utilizar proceso alternativo%
        if Mdefor2(k,7)==0;
            Mdefor2(k,11)=Mbond2(k,2)/Mbond2(k-1,11);
        end
    end
end

```

```

Mdefor2(k,12)=Mbond2(k,3)/Mbond2(k-1,12);
Mdefor2(k,10)=Mdefor2(k,12)/const2(k-1,1);
Mdefor2(k,4)=PM2(k,4)/(3*GM);
Mdefor2(k,9)=Mdefor2(k,10)-Mdefor2(k,4);
Mdefor2(k,8)=Mdefor2(k,9)/((2*alf*pppo2(k,2))/(M^2*(2*pppo2(k,1)+ks*s-
pppo2(k,3)))));
Mdefor2(k,7)=Mdefor2(k,11)/(const2(k-1,1)*const2(k-1,4));
Mdefor2(k,5)=Mdefor2(k,11)/const2(k-1,1);
Mdefor2(k,6)=Mdefor2(k,7)-Mdefor2(k,5);
Mdefor2(k,1)=Mdefor2(k,5)-Mdefor2(k,8);
Mdefor2(k,2)=Mdefor2(k,6);
Mdefor2(k,3)=Mdefor2(k,1)+Mdefor2(2,2);
Mmvol2(k,1)=Mmvol2(k-1,1)-Mdefor2(k,5)*Mmvol2(k-1,1);
Mmvol2(k,2)=Mmvol2(k-1,2)-Mdefor2(k,6)*Mmvol2(k-1,2);
else
%Pasol7: Deformaciones plásticas volumétricas%
if Mdefor2(k,5)-Mdefor2(k,1)<0;
    Mdefor2(k,8)=0;
else
    Mdefor2(k,8)=Mdefor2(k,5)-Mdefor2(k,1);
end
%Pasol8: Deformaciones plásticas cortantes%
Mdefor2(k,9)=Mdefor2(k,8)*(2*alf*pppo2(k,2))/(M^2*(2*pppo2(k,1)+ks*s-
pppo2(k,3)));
%Pasol9: Deformación cortante total%
Mdefor2(k,10)=Mdefor2(k,4)+Mdefor2(k,9);
%Paso20: Deformaciones, esfuerzos y energía en el cemento%
Mdefor2(k,11)=Mdefor2(k,5)*const2(k-1,1);
Mdefor2(k,12)=Mdefor2(k,10)*const2(k-1,1);
end
Mbond2(k,6)=Mbond2(k,4)*Mdefor2(k,11);
Mbond2(k,7)=Mbond2(k,5)*Mdefor2(k,12);
Mbond2(k,8)=Mbond2(k,6)+Mbond2(k,7);
Mbond2(k,9)=Mbond2(k-1,9)+Mbond2(k,8);
%Paso21: Variable de daño, módulos y relaciones modulares%
if Cb==0;
    Mbond2(k,10)=0;
elseif log(Mbond2(k,9)/ro)/r1<Mbond2(k-1,10);
    Mbond2(k,10)=Mbond2(k-1,10);
else
    Mbond2(k,10)=log(Mbond2(k,9)/ro)/r1;
end;
Mbond2(k,1)=Mbond2(1,1)*exp(-Mbond2(k,10)/2);
Mbond2(k,11)=Mbond2(1,11)*exp(-Mbond2(k,10));
Mbond2(k,12)=Mbond2(1,12)*exp(-Mbond2(k,10));
if pppo2(k,1)>PMc;
    KMm2(k,1)=pppo2(k,1)*Mmvol2(k,1)/kappaM;
    KMm2(k,2)=(pppo2(k,1)+s)*Mmvol2(k,2)/kappaM;
    KMm2(k,3)=KMm2(k,1)*KMm2(k,2)/(KMm2(k,1)+KMm2(k,2));
else
    KMm2(k,1)=PMc*Mmvol2(k,1)/kappaM;

```

```

        KMm2(k,2)=(PMc+s)*Mmvol2(k,2)/kappam;
        KMm2(k,3)=(KMm2(k,1)*KMm2(k,2))/(KMm2(k,1)+KMm2(k,2));
    end
    const2(k,1)=Mbond2(k,1)/(1+Cb*Mbond2(k,1));
    const2(k,2)=Mbond2(k,12)/(3*GM);
    const2(k,3)=Mbond2(k,11)/KMm2(k,1);
    const2(k,4)=KMm2(k,3)/KMm2(k,1);
    %Paso22: Deformaciones externas%
    pqext2(k,3)=pqext2(k-1,3)+Mdefor2(k,7);
    pqext2(k,4)=pqext2(k-1,4)+Mdefor2(k,10);
    %Paso23: Se inicia la siguiente iteración%
    k=k+1;
    %Esfuerzos externos%
    pqext2(k,2)=pqext2(k-1,2)+Dqext2;
    pqext2(k,1)=pqext2(k-1,1)+(Dqext2/PTEE);
    %Esfuerzos en la matriz%
    PM2(k,4)=(pqext2(k,2)-(pqpo2(k-1,2)+Mbond2(k-1,5)*const2(k-1,1)*Cb))/
(1+const2(k-1,2)*const2(k-1,1)*const2(k-1,1)*Cb);
    PM2(k,1)=(1+const2(k-1,2)*const2(k-1,1)*const2(k-1,1)*Cb);
    PM2(k,2)=1+const2(k-1,3)*const2(k-1,1)*Cb*const2(k-1,1)*const2(k-1,4);
    PM2(k,3)=(3*PM2(k,2))/PM2(k,1);
    pqpo2(k,2)=pqpo2(k-1,2)+PM2(k,4);
    pqpo2(k,1)=pqpo2(k-1,1)+(PM2(k,4)/PM2(k,3));
    Mbond2(k,3)=const2(k-1,2)*const2(k-1,1)*PM2(k,4);
    Mbond2(k,2)=const2(k-1,3)*const2(k-1,1)*(PM2(k,4)/PM2(k,3));
    Mbond2(k,5)=Mbond2(k-1,5)+Mbond2(k,3);
    Mbond2(k,4)=Mbond2(k-1,4)+Mbond2(k,2);
    envol2(k,1)=M*pqpo2(k,1)+M*ks*s;
    if (pqpo2(k,2)>=envol2(k,1));
        disp('Se alcanzó la condición de falla');
        break;
    end;
end;
end;
end
for i=1:k-1;
Vvt2(i,1)=Mmvol2(i,1)+Mmvol2(i,2);
end
%Gráficas de los resultados%
%Curva de compresibilidad de la macro-estructura%
subplot(2,1,1);
semilogx(pqext2(2:k-1,1),Mmvol2(2:k-1,1));
title('Compresibilidad de la macro-estructura');
xlabel('Pext (MPa)');
ylabel('vM (Mpa)');
%Compresibilidad VM+Vm%
subplot(2,1,2);
semilogx(pqext2(2:k-1,1),Vvt2(2:k-1,1));
title('Curva de Compresibilidad total VM+vm');
xlabel('Pext (MPa)');
ylabel('vM+vm');

```

