```
%cjfun1a main  - 0.2 m mesh
%reads ntcf input file and calls a Carslaw and Jaeger analitical function
%reports back qh
%used first in MF test: John Walton's Case 1
%Carslaw and Jaeger pp.70-71
%8/29/02: increas number of time divisions to 140
clear all
%works for 25 yrs regime

%this is an emulation instead of a true input processing
%this emulation produces tcent (central), tp (perturbation), ts (time)

%Step 1: emulate ts
n1=500; %number of time divisions
quo=4^(1/n1);
tyr=[0 quo.^[1:n1]-1];
tyr(1)=tyr(2)/2;tyr=[0 tyr]; %insert a midpoint between zero and first time, 8/23/02
tyr(1)=tyr(2)/2;tyr=[0 tyr]; %insert a midpoint between zero and first time, 8/29/02
ts=tyr*3600*24*365.25; %sec
ts(1)=1e-12;

%Step 2: emulate tcent (it will be a Case 1 temp at 1 m distance)
k=2.0;
al=.85e-6;
hc=2.0;
h=hc/k;
V=(tyr*0+1)';
%V=(tyr/3)';
T(:,1)=V;
x=0;
[m n]=size(ts);
N=50;
for L=1:N
    V=T(:,L);
for i=1:n
    vi=0;
    for j=0:i-1
        if j==0
             Vj=V(1);
             t=ts(i);
        else
             Vj=V(j+1)-V(j);
             t=ts(i)-ts(j);
        end
        c=h*(al*t)^.5;
        d=h*x+h^2*al*t;
        e=x/(2*(al*t)^0.5);
        if d < 709
            vi=vi+Vj*(erfc(e) - exp(d).*erfc(e+c));%cooling
        else %exp(s7)*erfc(s6) will be replaced with first Taylor term:
            'limit exceeded'
        end
    end
        v(i,1)=vi;
end
%heat flux density
q=(V-v)*hc;
dV=-q./(0.1*1000);
if L==50
    Q=q;
    tw=v;
    ta=V;
    %subplot(2,1,1), plot(tyr,tw)
    %subplot(2,1,2), plot(tyr,q)
end

T(:,L+1)=T(:,L)+dV;
vv(:,L)=v;
end

%extract tcent from vv
M=30; %number of intervals -- one more will be added
```

```
t3=vv(:,50); %use 3rd as base for equidistant divison in temperature
dtn=-(t3(2)-t3(n))/(M-1);
tempb=t3(2):dtn:t3(n);
timeb=interp1(t3(2:n)',ts(2:n),tempb,'lin');
timeb(M+1)=timeb(M);
timeb(M)=(timeb(M)+timeb(M-1))/2/1.05;

t1=vv(:,1);  %surface temperature of L=1 m, Repeat for L=250 m and L=500

%here modify the base time division 8/16/02
timeb=[timeb(1:26) (1:.25:3)*3600*24*365.25];

%insert midpoint in each time division, 8/23/02
timea=(timeb(1:34)+timeb(2:35))/2;
time(1:2:70)=timeb;
time(2:2:68)=timea;
timeb=time;
%insert midpoint in each time division, 8/29/02
timea=(timeb(1:34)+timeb(2:35))/2;
time(1:2:70)=timeb;
time(2:2:68)=timea;
timeb=time;

tcent=interp1(ts(2:n),t1(2:n),timeb);

%reduce analytical to timeb
%for L=1:500
%ttgl_cj_red(:,L)=interp1(ts(2:n), vv(2:n,L)',timeb','lin');
%ttal_cj_red(:,L)=interp1(ts(2:n), T(2:n,L)',timeb','lin');
%end

%hold off
% plot(timeb,tcent,'.-')
% hold on
% plot(ts,t1,ts,t3,'--')
% hold off

%make TT matrix from tcent
[m2 n2]=size(tcent);

TT=zeros(n2);
for i=1:n2
    ii=[1:i];
    TT(n2-ii+1,n2-i+1)=tcent(n2-ii+1)';
end
TT=TT+triu(TT',1);

%process qh wall heat flux for each TT column
tb=timeb;
for N=1:n2
    %use the 1st-kind boundary solution
    V=TT(:,N);
    for i=1:n2
        qi=0;
        for j=0:i-2
            if j==0
                 Vj=V(1);
                 t_2=tb(i);
                 t_1=tb(i-1);
            else
                 Vj=V(j+1)-V(j);
                 t_2=tb(i)-tb(j);
                 t_1=tb(i-1)-tb(j);
            end
            c=2*(k/(pi*al)^.5)/(t_2^.5+t_1^.5);
            qi=qi+Vj*c;%cooling
        end
        j=i-1; %last step -- diagonal element
            if j==0
              Vj=V(1);
                 t_2=tb(i);
                 t_1=0;
```

```
            else
                 Vj=V(j+1)-V(j);
                 t_2=tb(i);
                 t_1=tb(i-1);
            end
            t=t_2-t_1;
            c=2*(k/(pi*al*t)^.5); %use 2 for average value of a 1/sqrt(tb) integral
            qi=qi+Vj*c;%cooling
        qq(i,1)=qi;
     end
    qh(:,N)=qq;
end

%calculate CJ rock model NTCF matrix hh
hh=qh/TT;
save 'hh.dat'  hh -ascii  %repeat for L=250 m, and L=500 m
save 'qh'  qh -ascii
save 'T'  TT -ascii
save 'tcent.dat'  tcent -ascii
save 'time.dat'  time -ascii
```

Intentionally Left Blank