```
---------------------
rk_inte1.instructions
---------------------

- rk_inte1.pro:
  rk4-integration of 2-body orbit
  for given orbital elements and time interval
  checks the conservation of elements
                -"-          angular momentum L
                -"-          energy E
  plots the orbit + analytial solution


- prints the following instruction when called without parameters:

  IDL> rk_inte1


-----------------------------------------------------------
pro rk_inte1,elem,t1,t2,dt,output=output,myy=myy,plot=plot
-----------------------------------------------------------

Cartesian integration of non-perturbed 2-body orbit HS 20.11.02
 elem=[a,e,i,ome,w,tau]  initial orbital elements
 t1,t2                   integration time interval (orbital periods)
```

```
 dt                        time step (orbital periods)
KEYWORDS:
 taylor                    use Taylor series, with degree=taylor (def=rk4)
 output                    output interval of ELEM,L,E in steps (def=nsteps/10)
                           first line -> original values of L and E
                           output negative -> just store
 t_out,l_out,e_out         dL/L and dE/e vs t_out (stored every |output| step)
 myy                       G * (m1+m2)   def=1.
 /example                  example of integration:
                              a=1,ecc=0.5,i=10,ome=90.,w=0,tau=0
                    t1=0, t2=10*TORB, dt=0.01*TORB
 plot=istep                plot every istep steps
 wid                       limit of plot region
 /cplot                    plot analytic solution (white squares)
  oplot=color              plot on top of previous orbit with
                           color=oplot+2 (i.e 1->col=3=green)
 dl,de                     return averaed change in dL/L and dE/E
                                  /orbit period

EXAMPLE:
 rk_inte1,elem,t1,t2,dt,/example,/plot
-----------------------------------------------------------

---------------------------------
;1) Start by trying the example values
```

```
;--------------------------------------------------

    rk_inte1,elem,t1,t2,dt,/example,/plot

; -plots an orbit integrated for 10 periods
; -prints a table of orbital elements at different times
;  as well as dL/L, dE/E

; -and has also returned variables:
;  to check what are the values of orbital elements
;  eccentricity should be 0.5

    print,elem            ;elem=[a,e,i,ome,w,tau]
    print,t1,t2,dt        ;in orbital periods

;Why does the orbit look like it does?

;--------------------------------------------------
;  same integrated orbit with small white squares=analytical orbit

    rk_inte1,elem,t1,t2,dt,/example,/plot,/cplot
```

```
;---------------------------------------------------------------------------
;2) Try now the same orbital elements (omit /example from the call)
;   with larger timesteps (The previous example has returned dt=0.01)
;   with oplot=1,2,3,... orbits from different calls are plotted on the same plot


     rk_inte1,elem,t1,t2,0.01,/plot,/cplot

     rk_inte1,elem,t1,t2,0.02,/plot,oplot=1

     rk_inte1,elem,t1,t2,0.03,/plot,oplot=3  ;starts to look bad?

     rk_inte1,elem,t1,t2,0.05,/plot,oplot=4  ;escape!


;remember: to get rid of windows use wide

     wide

;------------------------------------------------------------------------------
;3) The integration thus gets progressively more inacurate with increasing time-step
;   Let's look how the error increases with time
;   t_out, e_out, l_out keywords return   dL/L and dE/E vs. time
;   output determines storing interval in steps
```

```
;   negative -> does not print to terminal

;3a) A short time interval (three orbits),
;    storing dL/L and dE/E at every step

     rk_inte1,elem,0.0,3.0,0.001,/plot,t_out=t_out,e_out=e_out,l_out=l_out,output=-1

     !p.multi=[0,2,1]
     nwin
     plot,t_out,e_out,psym=-4,xtitle='T/PER',ytitle='dE/E',title='dt=0.001 EKS=0.5'
     plot,t_out,l_out,psym=-4,xtitle='T/PER',ytitle='dL/L'
     !p.multi=0

;On what part of the orbit  does the error mainly occur?

;3b) Take a longer interval


     rk_inte1,elem,0.,10,0.01,/plot,t_out=t_out,e_out=e_out,l_out=l_out,output=-20

     !p.multi=[0,2,1]
     nwin
     plot,t_out,e_out,psym=-4,xtitle='T/PER',ytitle='dE/E',title='dt=0.01,EKS=0.5'
     plot,t_out,l_out,psym=-4,xtitle='T/PER',ytitle='dL/L'
     !p.multi=0
```

```
;So, the overall error seems to increase linearly with time
;However, this is true only when the orbit does not change too much
;try larger dt=0.05

     rk_inte1,elem,0.,10,0.05,/plot,t_out=t_out,e_out=e_out,l_out=l_out,output=4

     !p.multi=[0,2,1]
     nwin
     plot,t_out,e_out,psym=-4,xtitle='T/PER',ytitle='dE/E',title='dt=0.05,EKS=0.5'
     plot,t_out,l_out,psym=-4,xtitle='T/PER',ytitle='dL/L'
     !p.multi=0

;why is there no accumulation of error in the end of this integration?

;-----------------------------------------------------------------------
;4) Let's plot the dL/L and dE/E (/per orbital period) versus timestep
;   These are returned by 'dl' and 'de' keywords
;   Here is an example of how to do it compactly, without writing a procedure
;   So, just copy lines with mouse and move to IDL window!


   dt_tab=[.01,.005,.0025,.001]*1.d0
   dl_tab=dt_tab
```

```
   de_tab=dt_tab

   for i=0,n_elements(dt_tab)-1 do begin &  rk_inte1,elem,t1,t2,dt_tab(i),plot=1,oplot=i,dl=dl,d

   nwin
   plot,dt_tab,abs(dl_tab),psym=-4,xtit='DT',ytit='dE/E, dL/L (/orbit)',title='RK4'
   oplot,dt_tab,abs(de_tab),psym=-6,col=2

;Error seems to increase very fast with time-step
;Try with log-log plot:

  nwin
  plot,dt_tab,abs(dl_tab),psym=-4,/xlog,/ylog,yr=[1d-15,1],xtit='DT',ytit='dE/E, dL/L (/orbit)',
  oplot,dt_tab,abs(de_tab),psym=-6,col=2

;Errors seem to behave as dE/E = a * dt^k (which implies log(dE/E) = log(a)+k * log(dt)
;Try to determine k, by overplotting various lines:

  oplot,dt_tab,dt_tab^2,col=3,lines=2
  oplot,dt_tab,dt_tab^3,col=5,lines=2
  oplot,dt_tab,dt_tab^4,col=6,lines=2
  oplot,dt_tab,dt_tab^5,col=7,lines=2
  oplot,dt_tab,dt_tab^6,col=8,lines=2

;Which curve seems to have the right slope?
```

```
  oplot,dt_tab,de_tab(0)*(dt_tab/dt_tab(0))^5,psym=-1,thick=3,sym=1,lines=2  ; almost perfect?

;How does the result agree with the RK4 being a fourth-order method?


;-----------------------------------
;5) So, errors in dL/L and dE/E depends on timestep
;   How does it depend on other things, like a and eks?


;5a) try different semimajor-axis:

  elem1=[1.0,  0.5,   0.,  0.,  0.,  0.] ;elem=[a,e,i,ome,w,tau]
  elem2=[2.0,  0.5,   0.,  0.,  0.,  0.] ;so a is changed from a=1 to a=2

  dt=0.01
  t1=0.
  t2=10.

  rk_inte1,elem1,t1,t2,dt,dl=dl1,de=de1,/plot,wid=4
  rk_inte1,elem2,t1,t2,dt,dl=dl2,de=de2,/plot,oplot=1           ;add also /plot to check the cha

; check the changes in E and L
; Is there any difference?
```

```
print,'error dE/E', de1,de2
print,'error dL/L', dl1,dl2




;5b) try different eccentricities:

  elem1=[1.0,  0.5,   0.,  0.,  0.,  0.] ;elem=[a,e,i,ome,w,tau]
  elem2=[1.0,  0.05,   0.,  0.,  0.,  0.] ;so eks is reduced form eks=0.5 to eks=0.05

  dt=0.01
  t1=0.
  t2=10.

  rk_inte1,elem1,t1,t2,dt,dl=dl1,de=de1,/plot,wid=4
  rk_inte1,elem2,t1,t2,dt,dl=dl2,de=de2,/plot,oplot=1           ;add also /plot to check the cha

; check the changes in E and L

  print,de1,de2
  print,dl1,dl2

; So, how does the eccentricity affect?
```

```
;5c) Let's try for a range of different values
;    choose dt=0.001 to avoid too large changes with eks=0.9!

   eks_tab=[.001,.01,.1,.25,.5,.75,.9,.95]*1.d0
   dl_tab=eks_tab
   de_tab=eks_tab
   t1=0.
   t2=10.
   dt=0.001
   output=10000   ; not to print so much

   for i=0,n_elements(eks_tab)-1 do begin &  rk_inte1,[1.,eks_tab(i),0.,0.,0.,0.],t1,t2,dt,plot=


   nwin
   plot,eks_tab,abs(dl_tab),/xlog,/ylog,psym=-4,xtitle='eccentricty',ytitle='error',yr=[1d-16,1.
   oplot,eks_tab,abs(de_tab),psym=-6,col=2

;so things get really bad when eccentricity is increased
;try to look it this way

   nwin
   plot,1.-eks_tab,abs(dl_tab),/xlog,/ylog,psym=-4,xtitle='1-eccentricty',ytitle='error',yr=[1d-
   oplot,1.-eks_tab,abs(de_tab),psym=-6,col=2
```

```
;approximative fit?
   apu=lindgen(100)*.01+.01
   oplot,apu,1d-12/apu^8,lines=2,col=3

;Remembering the result in 3a) how would you intepret this stong eccentricity dependence?


;just plotting the time evolution with EKS=0.9 and EKS=0.5
     elem1=[1.,0.9,0.,0.,0.,0.]
     rk_inte1,elem1,0.0,3.0,0.001,/plot,t_out=t_out1,e_out=e_out1,l_out=l_out1,output=-1

     elem2=[1.,0.5,0.,0.,0.,0.]
     rk_inte1,elem2,0.0,3.0,0.001,/plot,oplot=1,t_out=t_out2,e_out=e_out2,l_out=l_out2,output=-1

     !p.multi=[0,2,1]
     nwin
     plot,t_out1,e_out1,xtit='T/PER',ytit='dE/E',title='dt=0.001 EKS=0.9 (red) 0.1 (GREEN)',col=
     oplot,t_out2,e_out2,col=3
     plot,t_out1,-l_out1,xtit='T/PER',ytit='dL/L',col=2,/ylog,yr=[1d-12,1]
     oplot,t_out2,-l_out2,col=3
     !p.multi=0


;-------------------------------------------------------------------------------------
```

```
;6 Let's now have a look at errors in Taylor-series integration
;  keyword taylor defines the degree (1 or 2)



;6a) TAYLOR I

     elem=[1.,0.5,0.,0.,0.,0.]

     dt=0.01
     rk_inte1,elem,0.0,1.0,dt,/plot,output=-1,/cplot,taylor=1,title='Taylor I: dt=0.01, 0.001, 0

     dt=0.001
     rk_inte1,elem,0.0,1.0,dt,/plot,output=-1,taylor=1,oplot=1 ; bad

     dt=0.0001
     rk_inte1,elem,0.0,1.0,dt,/plot,output=-1,taylor=1,oplot=2 ; still bad, has not completed on

;6b) TAYLOR II

     elem=[1.,0.5,0.,0.,0.,0.]

     dt=0.01
     rk_inte1,elem,0.0,1.0,dt,/plot,output=-1,/cplot,taylor=2,title='Taylor 2: dt=0.01, 0.001, 0
```

```
    dt=0.001
    rk_inte1,elem,0.0,1.0,dt,/plot,output=-1,taylor=2,oplot=1 ; better

    dt=0.0001
    rk_inte1,elem,0.0,1.0,dt,/plot,output=-1,taylor=2,oplot=2 ; almost acceptable


;-------------------------------------------
;6c) Let's check the error vs. dt dependence, as we did for RK4

;with taylor II
;if this seems to take forever, remove the shortest time-step

   elem=[1.,0.5,0.,0.,0.,0.]
   dt_tab=[.01,.001,.0001,.00001]*1.d0
   dl_tab=dt_tab
   de_tab=dt_tab
   t1=0.d0
   t2=2.d0

   for i=0,n_elements(dt_tab)-1 do begin &  rk_inte1,elem,t1,t2,dt_tab(i),plot=0,oplot=i,dl=dl,d

  nwin
  plot,dt_tab,abs(dl_tab),psym=-4,/xlog,/ylog,yr=[1d-12,1],xtit='DT',ytit='dE/E, dL/L (/orbit)',
  oplot,dt_tab,abs(de_tab),psym=-6,col=2
```

```
;again, check the exponent of DE/E vs dt^k dependence

  oplot,dt_tab,1d3*dt_tab^2,lines=2
  oplot,dt_tab,1d3*dt_tab^3,lines=2

;Error proportional to dt^-3 ?

;------------------------------------
;Repeat the same with eks=0.05
;let's plot them on top of the previous error-curves

   elem2=[1.,0.05,0.,0.,0.,0.]
   dt_tab=[.01,.001,.0001,.00001]*1.d0
   dl_tab2=dt_tab
   de_tab2=dt_tab
   t1=0.d0
   t2=2.d0

   for i=0,n_elements(dt_tab)-1 do begin &  rk_inte1,elem2,t1,t2,dt_tab(i),plot=0,oplot=i,dl=dl,

;previous (eks=0.5) and current values (EKS=0.05)

  nwin
```

```
  plot,dt_tab,abs(dl_tab),psym=-4,/xlog,/ylog,yr=[1d-12,1],xtit='DT',ytit='dE/E, dL/L (/orbit)',
  oplot,dt_tab,abs(dl_tab2),psym=-1

  oplot,dt_tab,abs(de_tab),psym=-6,col=2
  oplot,dt_tab,abs(de_tab2),psym=-1,col=2

  oplot,dt_tab,1d3*dt_tab^2,lines=2
  oplot,dt_tab,1d3*dt_tab^3,lines=2


;So the errors increase with eccentricity, but the same form of time dependence is retained



;--------------------------------------------
;6d) Do the same with with taylor I
;if this seems to take forever, remove the shortest time-step

   elem=[1.,0.5,0.,0.,0.,0.]
   dt_tab=[.01,.001,.0001,.00001]*1.d0
   dl_tab=dt_tab
   de_tab=dt_tab
   t1=0.d0
   t2=2.d0
```

```
  for i=0,n_elements(dt_tab)-1 do begin &  rk_inte1,elem,t1,t2,dt_tab(i),plot=0,oplot=i,dl=dl,d

 nwin
 plot,dt_tab,abs(dl_tab),psym=-4,/xlog,/ylog,yr=[1d-12,1],xtit='DT',ytit='dE/E, dL/L (/orbit)',
 oplot,dt_tab,abs(de_tab),psym=-6,col=2


;again, check the exponent of DE/E vs dt^k dependence

 oplot,dt_tab,1d3*dt_tab^1,lines=2
 oplot,dt_tab,1d3*dt_tab^2,lines=2


;------------------------------------
;Repeat the same with eks=0.05
;let's plot them on top of the previous error-curves

  elem2=[1.,0.05,0.,0.,0.,0.]
  dt_tab=[.01d0,.001d0,.0001d0,.00001d0]*1.d0
  dl_tab2=dt_tab
  de_tab2=dt_tab
  t1=0.d0
  t2=2.d0

  for i=0,n_elements(dt_tab)-1 do begin &  rk_inte1,elem2,t1,t2,dt_tab(i),plot=0,oplot=i,dl=dl,
```

```
;previous (eks=0.5) and current values (EKS=0.05)

  nwin
  plot,dt_tab,abs(dl_tab),psym=-4,/xlog,/ylog,yr=[1d-5,1],xtit='DT',ytit='dE/E, dL/L (/orbit)',t
  oplot,dt_tab,abs(dl_tab2),psym=-1

  oplot,dt_tab,abs(de_tab),psym=-6,col=2
  oplot,dt_tab,abs(de_tab2),psym=-1,col=2

  oplot,dt_tab,1d3*dt_tab^1,lines=2
  oplot,dt_tab,1d3*dt_tab^2,lines=2
  oplot,dt_tab,1d3*dt_tab^0.5,lines=2

;So the errors increase with eccentricity, but the same form of time dependence is retained
;this error is proportional to dt, rather that dt^2

;HOW COULD THIS BE POSSIBLE?
```