Variables ODEA

real*8 integer logical real*4 character

A

- a/(3,3) ! semi-major axis or pericentric distance if parabola / intermediary to solve Kepler's equation for hyperbola/matrix to the invariant plane
- a0 ! intermediary to solve Kepler's equation for hyperbola
- a1 ! intermediary to solve Kepler's equation for hyperbola
- a11 ! intermediary to solve Kepler's equation for hyperbola
- a3 ! intermediary to solve Kepler's equation for hyperbola
- a5 ! intermediary to solve Kepler's equation for hyperbola
- a7 ! intermediary to solve Kepler's equation for hyperbola
- a9 ! intermediary to solve Kepler's equation for hyperbola
- abn ! abs(n)
- amax ! maximum semi-major axis for tp
- amaxx ! $amax^{xx+1}$
- amin ! minimum semi-major axis for tp
- aminx ! $amin^{xx+1}$
- angle
- au = 1.495978707 10^8 ! km
- axbbeg(nplmax) ! barycentric accelerations of bodies at beginning of step
- axbend(nplmax) ! barycentric accelerations of bodies at end of step
- axbttp ! intermediate barycentric acceleration of tp
- axjt(nptmax) ! hierarchical Jacobi accelerations of tp
- axjttp ! intermediate hierarchical Jacobi acceleration of tp
- aybbeg(nplmax) ! barycentric accelerations of bodies at beginning of step
- aybend(nplmax) ! barycentric accelerations of bodies at end of step
- aybttp ! intermediate barycentric acceleration of tp
- ayjt(nptmax) ! hierarchical Jacobi accelerations of tp
- ayjttp ! intermediate hierarchical Jacobi acceleration of tp
- azbbeg(nplmax) ! barycentric accelerations of bodies at beginning of step
- azbend(nplmax) ! barycentric accelerations of bodies at end of step
- azbttp ! intermediate barycentric acceleration of tp

- azjt(nptmax) ! hierarchical Jacobi accelerations of tp
- azjttp ! intermediate hierarchical Jacobi acceleration of tp

B

- b ! intermediary to solve Kepler's equation for hyperbola
- b1 ! intermediary to solve Kepler's equation for hyperbola
- b11 ! intermediary to solve Kepler's equation for hyperbola
- b3 ! intermediary to solve Kepler's equation for hyperbola
- b5 ! intermediary to solve Kepler's equation for hyperbola
- b7 ! intermediary to solve Kepler's equation for hyperbola
- b9 ! intermediary to solve Kepler's equation for hyperbola
- biga ! semi-major axis or pericentric distance if parabola / intermediary to solve Kepler's equation for hyperbola
- bigb ! semi-major axis or pericentric distance if parabola / intermediary to solve Kepler's equation for hyperbola
- bot ! Intermediary to compute matrix to invariable plane

C

- c1 ! angular momentum element
- c2 ! angular momentum element
- c3 ! angular momentum element
- ca ! cos(x+m)
- cape ! eccentric anomaly
- capf ! eccentric anomaly for hyperbola
- capm ! Mean anomaly
- capn ! hyperbola mean anomaly
- capom ! Longitude of the ascending node
- ccap ! cos(cape)
- cen ! True if body is a satellite
- change ! True if a hierarchy change occurred at that step
- chcap ! ch(capf)
- checkchange ! True if the hierarchy should be checked
- chx ! ch(x)
- ci ! cos(inc)
- cimax ! cos(imax)
- cm ! cos(m)

- co ! cos(capom)
- cp ! cos(omega)
- cx ! cos(x)
- cw ! cos(w)

D

- d11 ! cp*co - sp*so*ci $\cos\omega\cos\Omega - \sin\omega\sin\Omega\cos i$
- d12 ! cp*so + sp*co*ci $\cos\omega\sin\Omega + \sin\omega\cos\Omega\cos i$
- d13 ! sp*si $\sin\omega\sin i$
- d21 ! -sp*co - cp*so*ci $-\sin\omega\cos\Omega - \cos\omega\sin\Omega\cos i$
- d22 ! -sp*so + cp*co*ci $-\sin\omega\sin\Omega + \cos\omega\cos\Omega\cos i$
- d23 ! cp*si $\cos\omega\sin i$
- dataname(80) ! output of the file operation (cp)
- de
- danbyac = 10^{-14} ! convergence criteria for danby
- danbyb = 10^{-14} ! convergence criteria for danby
- degrad = 180/pi
- dkloc ! disk plane (0 -> ecliptic, 1 -> centers, 2 -> invariable, 3 -> other plane, -n -> neme orbit)
- diff ! Rest if non convergence of Kepler equation for hyperbola
- dirs(80) ! directory path (usually .../workdir)
- diro(80) ! data path (usually path/run_1)
- dplfile ! dump file name
- dr = 1.7453292519943294 10^{-2} ! π/180
- dt ! time step
- dtdump ! time between dumps
- dth ! middle time step
- dtout ! time between binary output
- dx ! intermediary to get eccentric anomaly from e and m

E

- e ! eccentricity
- ec ! e cos(x)
- eca ! e cos(x+m)
- ech ! e ch(x)

- eltot(3) ! total angular momentum vector in the barycentric ref
- elx ! Component of an angular momentum
- ely ! Component of an angular momentum
- elz ! Component of an angular momentum
- em1 ! e-1
- emax ! maximum eccentricity for tp
- emin ! minimum eccentricity for tp
- energy ! total energy
- energy0 ! initial total energy
- eoff ! energy offset = 0 ?
- es ! e sin(x)
- esa ! e sin(x+m)
- esh ! e sh(x)
- eta (nplmax) ! masses of the orbits centers
- etatp(ntpmax) ! orbit mass of tp orbit
- etatpt ! orbit mass of tp orbit

F

- fachill ! Multiple of Hill radius
- f ! intermediary to get eccentric anomaly from e and m
- fac ! sqrt(hx**2 + hy**2)/h
- face ! intermediate to compute eccentric anomaly
- fname(80) ! file name
- fopenstat(80) ! file access (new, append, unknown)
- format(80) ! file format
- fp ! intermediary to get eccentric anomaly from e and m
- fpp ! intermediary to get eccentric anomaly from e and m
- fppp ! intermediary to get eccentric anomaly from e and m
- fverb ! verbose frequency (in terms of tout)

G

- genfile(80) ! name of gen file (gen_plae_hjs.sh usually)
- gm ! G * total mass of the orbit
- gmsum ! G * total mass of the orbit
- gname(80) ! data subdirectory (usually run_1)

H

- h2 ! squared norm of the angular momentum
- h ! norm of the angular momentum
- hx ! component of the angular momentum
- hy ! component of the angular momentum
- hz ! component of the angular momentum

I

- i ! variable in for loops (iterations, bodies)
- i1st ! 0 if first time, 1 otherwise
- i1sttp
- ialpha ! Conic section type (sign of e-1)
- icflg ! Coordinate menu (0 -> Ecliptic, 1-> Invariable plane)
- ierr ! iostat, problem of file opening if not zero
- ii ! counter for tp
- id ! Number of the orbit
- idisk ! Disk inclination
- iflag ! 1 if m > pi / 1 if n<0 / 1 if q<0
- iflgchk ! (write real*4 binary file, write real*8 binary file rather that real*4 ignored if bit0=T, calc energy of system wrt time, calc jacobi of the test particles, check if particles are removed, include J2 and J4 terms, initial hierarchy check, adaptable time step)
- in ! Number of the orbit
- init ! 0 if initialize, 1 otherwise
- iverb ! counter for tout
- imax ! maximum number of iteration/maximum inclination of tp
- inc ! inclination
- inparfile(80) ! name of parameter data file (usually paramshjs.in)
- inplfile(80) ! name of planet data file (usually plhjs.in)
- intpfile(80) ! name of test particle data file (usually tphjs.in)
- io_nbits ! number of bytes in iflgchk
- ir3jbeg(nplmax)
- ir3jend(nplmax)
- iread ! Number of the frame
- irem ! counter for discarded tp

- irflg ! Planetary radius menu (0 -> do not include radius, 1-> include the physical radius, 2 -> include a multiple of the Hill radius)
- irij3
- iseed ! for random function
- istat(ntpmax,nstat) ! tp status flag (0 if active, 1 if discarded ; if non active then -1 for not convergence of Danby, -2 if a<0 and r>rmax, -3 if r>rmax,-4 if q<qmin,1 if r<rmin, n if too close to body n)
- istati(nstat)
- istold(ntp) ! tp status flag (only active information, 0 if active, 1 if discarded)
- iu ! unit number to open file
- iuf ! unit number for frame file
- iud ! unit number for dump file
- iue ! unit number for energy file
- iuo ! unit number for oloc file (hierarchy)
- iuotp ! unit number for oloctp file (hierarchy tp)
- iuflg ! Units menu (0 -> solar masses and au, 1 -> au and year)
- iwrite ! 0 if file was never writte, 1 otherwise
- izero (nstat) ! zero array the same size as istati

J

- j ! variable in for loops (iterations, bodies, orbits)
- jj ! counter for bodies or orbits

K

- k ! variable in for loops (orbits)
- ke ! kinetic energy in the barycentric energy
- kk ! counter for orbits

L

- lclose ! True if bodies have radius

M

- m ! mean anomaly
- mass(nplmax)/(nbod) ! mass of bodies
- masstemp(nplmax) ! total mass for tp centers
- mat(nplmax,nplmax) ! tranform matrix from barycentric to hierarchical Jacobi
- mati(3,3) ! matrix to the invariant plane
- matr(3,3) ! transform matrix from the disk plane

- matp(nplmax,ntpmax) ! tranform matrix from barycentric to hierarchical Jacobi for tp
- matpt(nplmax) ! intermediate tranform matrix from barycentric to hierarchical Jacobi for tp
- mu(nplmax) ! masses of the orbits satellites
- mtot ! total mass of the system

N

- n ! hyperbola mean anomaly
- name(32)
- nbod ! number of massive bodies
- nc ! number of centers for tp
- niter ! number of the iteration
- nlag1 = 50 ! loop limits in the Laguerre attempts
- nlag2 = 400 ! loop limits in the Laguerre attempts
- nleft ! number of tps left
- nmax = 3 ! maximum iteration in Danby's quartic (Kepler equation if eccentric)
- nnn
- nper ! angle/twopi
- nplmax = 11 ! max number of bodies
- nptmax = 200,000 ! max number of test particles
- nsta ! number of dt for each time step
- nstat = nstatp + nplmax -1 ! Size of the test particle integer status flag
- nstatp = 3 ! number of status parameters
- nstatr = nstat ! Size of the test particle integer status flag
- ntp ! Number of tp for each orbit
- ntpt ! total number of tp

O

- odisk ! disk longitude of nodes
- ok ! False if any kind of problem (not good input data, hierarchy)
- okc ! False when no more particles to initialize
- oloc(nplmax,nplmax) ! hierarchy matrix (1 sat, -1 cen, 0 outer body)
- olocold(nplmax,nplmax) ! previous hierarchy matrix
- oloct(nplmax,ntpmax) oloc corresponding to the tp centers
- oloctt(nplmax) ! intermediate oloc corresponding to the tp centers

- orbct(nplmax) ! centers for tp
- omega ! longitude of periastron
- outfile(80) ! name of binary output file (usually bin_1)

P

- period
- pi = 3.14159265358979
- pi3by2 = 1.5 pi
- piby2 = pi/2
- pot ! potential energy

Q

- q ! parabola mean anomaly
- qmin ! smallest perihelion distance, <0 -> don't check

R

- r2hill(nplmax)
- rep(1)
- ri ! inverse of the radius
- rij2 ! rij**2
- rmax ! max distance from Sun, <0 -> don't check
- rmaxu ! max distance from Sun if not bound, <0 -> don't check
- rmin ! min distance from Sun, <0 -> don't check
- rpl(nplmax) ! radius of bodies
- rplsq(nplmax) ! radius of bodies, squared
- rr2 ! $(rbi-rbj)^2$
- rstat(ntpmax,nstatr) ! tp status flag : (time of the discard,perihelion distance if checked,encounter characteristics in rmvs)
- rstati(nstatr)
- rzero(nstatr) ! zero array the same size as rstati
-

S

- s ! random number
- sa ! sin(x+m)
- sat ! True if body is a satellite
- scap ! sin(cape)

- seed
- shcap ! sh(capf)
- si ! sin(inc)
- sm ! sin(m)
- smassyr = twopi * twopi
- so ! sin(capom)
- sp ! sin(omega)
- sqgma ! sqrt(gma)
- sqe ! $\sqrt{|1-e|}$
- sx ! sin(x)
- sw ! sin(w)
- shx ! sh(x)

T

- t ! time
- t0 ! initial time
- tdump ! time for next dump
- tfrac ! fraction done (t-t0)/(tstop-t0)
- time
- tout ! time for next output
- tstop ! final time
- test
- tiny = 4.10^{-15}
- tmp ! intermediary to solve Kepler's equation for hyperbola
- tmpf ! intermediary to compute eccentric anomaly for hyperbola
- twopi = 2 pi

U

- u ! argument of the latitude
- umat(nplmax,nplmax) ! inverse transform matrix from hierarchical Jacobi to barycentric
- umatp(nplmax,ntpmax) ! inverse transform matrix from hierarchical Jacobi to barycentric for tp
- umatpt(nplmax) ! intermediate inverse transform matrix from hierarchical Jacobi to barycentric for tp
- umpart(nplmax,nplmax) ! submatrix from the umat matrix for tp

V

- v ! norm of the velocity
- v2 ! squared norm of the velocity
- vdotr ! scalar product between velocity and position
- ver_num : 2 ! swift version
- vcen ! mass ratio -mu/(mu+eta)
- vfac1 ! Keplerian velocity
- vfac2 ! Keplerian velocity
- vsat ! mass ratio eta/(mu+eta)
- vx
- vxb(nplmax) ! barycentric velocities of bodies
- vxbtemp(nplmax) ! barycentric velocities of tp centers to compute centers plane
- vxbttp ! intermediate barycentric velocities for tp
- vxh(nplmax)
- vxj(nplmax) ! hierarchical Jacobi velocities of bodies
- vxjh(nplmax) ! middle hierarchical Jacobi velocities of bodies
- vxjt(nplmax) ! hierarchical Jacobi velocities of test particles
- vxjtemp(nplmax) ! hierarchical Jacobi velocities of tp centers to compute centers plane
- vxt ! intermediate velocity
- vy
- vyb(nplmax) ! barycentric velocities of bodies
- vybtemp(nplmax) ! barycentric velocities of tp centers to compute centers plane
- vybttp ! intermediate barycentric velocities for tp
- vyh(nplmax)
- vyj(nplmax) ! hierarchical Jacobi velocities of bodies
- vyjh(nplmax) ! middle hierarchical Jacobi velocities of bodies
- vyjt(nplmax) ! hierarchical Jacobi velocities of test particles
- vyjtemp(nplmax) ! hierarchical Jacobi velocities of tp centers to compute centers plane
- vyj
- vyt ! intermediate velocity
- vz
- vzb(nplmax) ! barycentric velocities of bodies
- vzbtemp(nplmax) ! barycentric velocities of tp centers to compute centers plane
- vzbttp ! intermediate barycentric velocities for tp

- vzh(nplmax)
- vzj(nplmax) ! hierarchical Jacobi velocities of bodies
- vzjh(nplmax) ! middle hierarchical Jacobi velocities of bodies
- vzjt(nplmax) ! hierarchical Jacobi velocities of test particles
- vzjtemp(nplmax) ! hierarchical Jacobi velocities of tp centers to compute centers plane
- vzt ! intermediate velocity

W

- w ! u - omega

X

- x/(nbod) ! coordinate / angle between 0 and 2π / approximate of the eccentric anomaly
- xb(nplmax) ! barycentric coordinates of bodies
- xbbeg(nplmax) ! barycentric coordinates of bodies at beginning of step
- xbend(nplmax) ! barycentric coordinates of bodies at end of step
- xbtemp(nplmax) ! barycentric coordinates of tp centers
- xbttp ! intermediate barycentric coordinates for tp
- xfac1 ! Keplerian coordinate
- xfac2 ! Keplerian coordinate
- xj(nplmax) ! hierarchical Jacobi coordinates of bodies
- xjbeg(nplmax) ! hierarchical Jacobi coordinates of bodies at beginning of step
- xjend(nplmax) ! hierarchical Jacobi coordinates of bodies at end of step
- xjo(nplmax) ! hierarchical Jacobi positions of bodies, possibly in the invariant plane
- xjt(ntpmax) ! hierarchical Jacobi coordinates of test particles
- xjtemp(nplmax) ! hierarchical Jacobi coordinates of tp centers
- xt ! intermediate position
- xx ! power index (a^xx has linear prior)/ Component of rbij

Y

- y/(nbod) ! coordinate
- yb(nplmax) ! barycentric coordinates of bodies
- ybbeg(nplmax) ! barycentric coordinates of bodies at beginning of step
- ybend(nplmax) ! barycentric coordinates of bodies at end of step
- ybtemp(nplmax) ! barycentric coordinates of tp centers
- ybttp ! intermediate barycentric coordinates for tp
- yj(nplmax) ! hierarchical Jacobi coordinates of bodies

- yjbeg(nplmax) ! hierarchical Jacobi coordinates of bodies at beginning of step
- yjend(nplmax) ! hierarchical Jacobi coordinates of bodies at end of step
- yjtemp(nplmax) ! hierarchical Jacobi coordinates of tp centers
- yjt(ntpmax) ! hierarchical Jacobi coordinates of test particles
- yt ! intermediate position
- yy ! Component of rbij

z

- z/(nbod) ! coordinate
- zb(nplmax) ! barycentric coordinates of bodies
- zbbeg(nplmax) ! barycentric coordinates of bodies at beginning of step
- zbend(nplmax) ! barycentric coordinates of bodies at end of step
- zbtemp(nplmax) ! barycentric coordinates of tp centers
- zbttp ! intermediate barycentric coordinates for tp
- zj(nplmax) ! hierarchical Jacobi coordinates of bodies
- zjbeg(nplmax) ! hierarchical Jacobi coordinates of bodies at beginning of step
- zjend(nplmax) ! hierarchical Jacobi coordinates of bodies at end of step
- zjt(ntpmax) ! hierarchical Jacobi coordinates of test particles
- zjtemp(nplmax) ! hierarchical Jacobi coordinates of tp centers
- zpara ! eccentric anomaly for parabola
- zt ! intermediate position
- zz ! Component of rbij