

```

0001 // ++++++
0002 funcprot(0);
0003 // ++++++
0004
0005 cv=%pi/180; // From degree to radian
0006
0007 // ++++++
0008 // ++++++
0009
0010 Latitude=52; // [°]
0011
0012 L=30; // Length of pendulum
0013 M=1;
0014
0015 // Initial data
0016
0017 alpha=45; // [°]
0018 beta=0;
0019 // [°] Initial position of pendulum
0020
0021 v0v=0.5;
0022 // Initial velocity
0023
0024 x0=[ L*sin(alpha*cv)*cos(beta*cv);..
0025      L*sin(alpha*cv)*sin(beta*cv);..
0026      -L*cos(alpha*cv)];
0027 // Initial pendulum position
0028
0029 // disp(x0);
0030
0031 v0=[ -v0v*cos(alpha*cv)*cos(beta*cv);..
0032      -v0v*cos(alpha*cv)*sin(beta*cv);..
0033      -v0v*sin(alpha*cv)];
0034 // Initial pendulum velocity
0035
0036 // disp(v0);
0037
0038 t0=0;
0039 tend=1800; // 180
0040 steps=1800; // 800
0041
0042 // ++++++
0043 // ++++++
0044
0045 // phi1 = geodesic latitude
0046
0047 phi1=Latitude*cv;
0048
0049 // ++++++
0050 // ++++++
0051
0052 // ω (radians/s) = earth's angular spin rate
0053
0054 omega=7.292115E-5;
0055 // omega=0;
0056
0057 // #####
0058
0059 // vector of coriolis acceleration
0060 // phi1 = geodesic (geographic) latitude
0061
0062 function rbc=fbC(v)

```

```

0063     v1=v(1); v2=v(2); v3=v(3);
0064
0065     sn=sin(phi1); cs=cos(phi1)
0066
0067     rbc(1)= 2*omega*(-v2*sn+v3*cs);
0068     rbc(2)= 2*omega*( v1*sn);
0069     rbc(3)=-2*omega*(-v1*cs);
0070 endfunction
0071
0072 // #####
0073
0074 function rbG=fbG()
0075     rbG(1)=0;
0076     rbG(2)=0;
0077     rbG(3)=-9.80616; // g45
0078 endfunction
0079
0080 // #####
0081 // #####
0082
0083 function rvDot=fvDot(x, v)
0084     // return ph data
0085
0086     x1=x(1); x2=x(2); x3=x(3);
0087     v1=v(1); v2=v(2); v3=v(3);
0088
0089     r=sqrt(x1^2+x2^2+x3^2);
0090
0091     ve12=v1^2+v2^2+v3^2;
0092
0093     xu=x/r;
0094
0095     bGC=fbG()+fbc(v);
0096
0097     bGCDOTxu=bGC(1)*xu(1)+bGC(2)*xu(2)+bGC(3)*xu(3);
0098
0099     bZWv=bGCDOTxu+(ve12/r);
0100
0101     bZW=-bZWv*xu;
0102
0103     b=bGC+bZW;
0104
0105     rvDot(1:3)=v;
0106     rvDot(4:6)=b;
0107 endfunction
0108
0109 // +++++
0110
0111 function rPendulum=fPendulum(t, ph)
0112     // returns 'phDot data'
0113     x=ph(1:3);
0114     v=ph(4:6);
0115     rPendulum=fvDot(x,v);
0116 endfunction;
0117
0118 // +++++
0119
0120 ph0(1:3)=x0;
0121 ph0(4:6)=v0;
0122
0123 tData=linspace(t0,tend,steps);
0124

```

```

0125 // ++++++
0126
0127 phData=ode(ph0,t0,tData,fPendulum);
0128
0129 // disp(phData);
0130
0131 // ++++++
0132
0133 to_deg=180/%pi;
0134
0135 // ++++++
0136
0137 PhiData=[];
0138
0139 nbpoints=size(phData,"c");
0140
0141 for Ix=1:nbpoints
0142     PhiData(Ix)=atan(phData(2,Ix)/phData(1,Ix));
0143
0144     // disp(PhiData(Ix));
0145 end
0146
0147 RadData=[];
0148
0149 for Ix=1:nbpoints
0150     RadData(Ix)=sqrt(phData(1,Ix)^2+phData(2,Ix)^2+phData(3,Ix)^2);
0151
0152     // disp(RadData(Ix));
0153 end
0154
0155 // disp(RadData);
0156
0157 // ++++++
0158
0159 my_handle=scf(1);
0160 clf(my_handle,"reset");
0161
0162 a=gca();
0163
0164 // ++++++
0165
0166 //drawlater();
0167 //
0168 //xtitle ( " Radius der Pendelbewegung " ,..
0169 //         " Anzahl der Integrationssschritte", ..
0170 //         " Radius " );
0171 //
0172 //a.labels_font_size=3;
0173 //a.x_label.font_size = 3;
0174 //a.y_label.font_size = 3;
0175 //a.title.font_size = 3;
0176 //
0177 //plot2d(phData(1,:)', phData(3,:)' );
0178 //
0179 //drawnow();
0180
0181 // ++++++
0182
0183 drawlater();
0184
0185 xtitle ( " Radius der Pendelbewegung " ,..
0186         " Anzahl der Integrationssschritte", ..

```

```
0187         " Radius " );
0188
0189 a.labels_font_size=3;
0190 a.x_label.font_size = 3;
0191 a.y_label.font_size = 3;
0192 a.title.font_size = 3;
0193
0194 plot(tData,RadData);
0195
0196 drawnow();
0197
0198 // ++++++
0199
0200 //drawlater();
0201 //
0202 //xtitle ( " Azimuthwinkel der Pendelbewegung " ,..
0203 //         " Anzahl der Integrationschritte", ..
0204 //         " Grad " );
0205 //
0206 //a.labels_font_size=3;
0207 //a.x_label.font_size = 3;
0208 //a.y_label.font_size = 3;
0209 //a.title.font_size = 3;
0210 //
0211 //plot(tData,PhiData*to_deg);
0212 //
0213 //
0214 //drawnow();
0215
0216 // ++++++
0217 // ++++++
```