

```

program movimiento_en_campo_gravitatorio_3;
{Movimientos de dos soles id,nticos y planetas en torno a ellos dos}
{Tierra y Marte}
{suma vectorial de dos aceleraciones}
{tiempo aconsejable const t=500 o menor}
uses
  Crt, Graph;
const
  t=1000;          {constantes generales}
  pi=3.1415926;
  g=6.67392E-11;   {6.67392 IAC}
  mA=1.9891E+30;   {1.9891E+30}
  mB=1.9891E+30;
  e=150/1.7E+11;   {factor de escala: hasta la orbita de Marte}
var
  GraphDriver : integer; { The Graphics device driver }
  GraphMode   : integer; { The Graphics mode value }
  MaxX, MaxY  : word;    { The maximum resolution of the screen }
  ErrorCode   : integer; { Reports any graphics errors }
  OldExitProc : Pointer; { Saves exit procedure address }

  asa,asax,asay,rsa0,rsax0,rsay0,vsax0,vsay0,physa0,psisa:real; {sol A}
  asb,asbx,asby,rsb0,rsbx0,rsby0,vsbx0,vsby0,physb0,psisb:real; {sol B}

  a,aa,ab,ax,ay,rpa0,rpax0,rpax0,rbp0,rpbx0,rpbx0,ry0,vx0,vy0,phy0,psi:real;

  a2,a2a,a2b,a2x,a2y,r2pa0,r2pax0,r2pay0,r2pb0,r2pbx0,r2pby0,r20,r2x0,r2y0,
  v2x0,v2y0,phy20,psi2:real; {planeta}
  rab:real;
  arsax0,arsay0,arsbx0,arsby0:real;
  axa,aya,axb,ayb:real;
  t:integer;
  ep,ec,tt:real;

  {$F+}
  procedure MyExitProc;
  begin
    ExitProc := OldExitProc; { Restore exit procedure address }
    CloseGraph;              { Shut down the graphics system }
  end; { MyExitProc }
  {$F-}

  procedure Inicia_modos_grafico;
  { Inicializa el modo grafico e informa de los errores que puedan ocurrir}
  var
    InGraphicsMode : boolean; { Flags initialization of graphics mode }
    PathToDriver    : string;  { Stores the DOS path to *.BGI & *.CHR }
  begin
    { when using Crt and graphics, turn off Crt's memory-mapped writes }
    DirectVideo := False;
    OldExitProc := ExitProc;          { save previous exit proc }
    ExitProc := @MyExitProc;          { insert our exit proc in chain }
  }
  PathToDriver := '';
  repeat

  {$IFDEF Use8514}          { check for Use8514 $DEFINE }

```

```

    GraphDriver := IBM8514;
    GraphMode := IBM8514Hi;
{$ELSE}
    GraphDriver := Detect;           { use autodetection }
{$ENDIF}

    InitGraph(GraphDriver, GraphMode, PathToDriver);
    ErrorCode := GraphResult;        { preserve error return }
    if ErrorCode <> grOK then         { error? }
    begin
        Writeln('Graphics error: ', GraphErrorMsg(ErrorCode));
        if ErrorCode = grFileNotFound then { Can't find driver file }
        begin
            Writeln('Enter full path to BGI driver or type <Ctrl-Break> to
quit:');
            Readln(PathToDriver);
            Writeln;
        end
        else
            Halt(1);                 { Some other error: terminate }
        end;
    until ErrorCode = grOK;
    MaxX := GetMaxX;                 { Get screen resolution values }
    MaxY := GetMaxY;
end; { Initialize }

```

```

procedure dibuja_sol(x,y,c:integer);
begin
    PutPixel(MaxX div 2 +x, MaxY div 2 +y, 1);
    SetLineStyle(SolidLn,0,NormWidth);setColor(c);
    Line(MaxX div 2 +x, MaxY div 2 +y+6,MaxX div 2 +x,MaxY div 2 +y-6);
    Line(MaxX div 2 +x-4,MaxY div 2 +y,MaxX div 2 +x+4,MaxY div 2 +y);
    PutPixel(MaxX div 2 +x+6, MaxY div 2 +y, c);
    PutPixel(MaxX div 2 +x-6, MaxY div 2 +y, c);
    PutPixel(MaxX div 2 +x, MaxY div 2 +9 +y, c);
    PutPixel(MaxX div 2 +x, MaxY div 2 -9 +y, c);
    PutPixel(MaxX div 2 +x+1, MaxY div 2 +y+1, c);
    PutPixel(MaxX div 2 +x+1, MaxY div 2 +y-1, c);
    PutPixel(MaxX div 2 +x-1, MaxY div 2 +y+1, c);
    PutPixel(MaxX div 2 +x-1, MaxY div 2 +y-1, c);
    PutPixel(MaxX div 2 +x+3, MaxY div 2 +y+3, c);
    PutPixel(MaxX div 2 +x+3, MaxY div 2 +y-3, c);
    PutPixel(MaxX div 2 +x-3, MaxY div 2 +y+3, c);
    PutPixel(MaxX div 2 +x-3, MaxY div 2 +y-3, c);
end;

procedure dibuja_los_ejes;
begin
    SetLineStyle(DottedLn,0,NormWidth);
    SetColor(4);
    Line(MaxX div 2,MaxY div 2 +230,MaxX div 2,MaxY div 2 -230);{
vertical }
    Line(MaxX div 2 -310,MaxY div 2,MaxX div 2 +310,MaxY div 2);{
horizontal }
    SetColor(8);ellipse(MaxX div 2, MaxY div 2,0,360,round(53*0.8),53);
    SetColor(6);ellipse(MaxX div 2, MaxY div 2,0,360,round(96*0.8),96);
    SetColor(1);ellipse(MaxX div 2, MaxY div
2,0,360,round(132*0.8),132);

```

```

        SetColor(4);ellipse(MaxX div 2, MaxY div
2,0,360,round(202*0.8),202);
        gotoxy(65,2);write('Escala:');
        TextColor(4);gotoxy(65,3);write('Marte');
        TextColor(3);gotoxy(65,8);write('Tierra');
        TextColor(6);gotoxy(65,10);write('Venus');
        TextColor(8);gotoxy(65,12);write('Mercurio');
    end;

procedure presentacion;
begin
    SetTextStyle(TriplexFont,0,3);SetTextJustify(CenterText,CenterText);
    SetColor(15);OutTextXY(320,70,'GRAVITY.EXE');
    SetColor(15);
    OutTextXY(540,410,'Pulsa Enter');
    Readln;ClearViewPort;
end;

procedure DatosIniciales;
begin
    rx0:=0;ry0:=1.5E+11;        {planeta1 azul: Tierra}
    r0:=sqrt(rx0*rx0+ry0*ry0);
    vx0:=-44500;vy0:=0;        {rango -43500 -45200 estable}

    r2x0:=0;r2y0:=2.27936E+11; {planeta2 rojo: Marte}
    r20:=sqrt(r2x0*r2x0+r2y0*r2y0);
    v2x0:=-34830;v2y0:=0;      {probar -34200 estable -33594 teorico}

    rab:=8E+10;
    rsax0:=-4E+10;rsay0:=0;    {posiciones iniciales sol A}
    rsbx0:=4E+10;rsby0:=0;    { sol B}
    vsax0:=0;vsay0:=-28804;    {28750 velocidades iniciales sol A}
    vsbx0:=0;vsby0:= 28804;    {calculado teorico sol B}
    physa0:=pi;                { angulo inicial sol A}
    physb0:=0;                { sol B}

end;

procedure calcula_phy;
begin
    if (ry0>0)and(rx0>0)then phy0:=arctan(ry0/rx0);
{planeta1}
    if (ry0>0)and(rx0<0)then phy0:=arctan(ry0/rx0)+pi;
    if (ry0<0)and(rx0<0)then phy0:=arctan(ry0/rx0)+pi;
    if (ry0<0)and(rx0>0)then phy0:=arctan(ry0/rx0);

    if (r2y0>0)and(r2x0>0)then phy20:=arctan(r2y0/r2x0);
{planeta2}
    if (r2y0>0)and(r2x0<0)then phy20:=arctan(r2y0/r2x0)+pi;
    if (r2y0<0)and(r2x0<0)then phy20:=arctan(r2y0/r2x0)+pi;
    if (r2y0<0)and(r2x0>0)then phy20:=arctan(r2y0/r2x0);

end;

procedure calcula_psi;
{ ngulos entre el planeta 1 y soles AyB y aceleraciones sufridas}
var psia,psib,axa,aya,axb,ayb:real;
begin
    { ngulo entre planeta y sol A}
    if ((ry0-rsay0)>0)and((rx0-rsax0)>0)then psia:=arctan((ry0-
rsay0)/(rx0-rsax0))+pi;
    if ((ry0-rsay0)>0)and((rx0-rsax0)<0)then psia:=arctan((ry0-
rsay0)/(rx0-rsax0));

```

```

        if ((ry0-rsay0)<0)and((rx0-rsax0)<0)then psia:=arctan((ry0-
rsay0)/(rx0-rsax0));
        if ((ry0-rsay0)<0)and((rx0-rsax0)>0)then psia:=arctan((ry0-
rsay0)/(rx0-rsax0))+pi;
        axa:=aa*cos(psia);aya:=aa*sin(psia);
        { ngulo entre planeta y sol B}
        if ((ry0-rsby0)>0)and((rx0-rsbx0)>0)then psib:=arctan((ry0-
rsby0)/(rx0-rsbx0))+pi;
        if ((ry0-rsby0)>0)and((rx0-rsbx0)<0)then psib:=arctan((ry0-
rsby0)/(rx0-rsbx0));
        if ((ry0-rsby0)<0)and((rx0-rsbx0)<0)then psib:=arctan((ry0-
rsby0)/(rx0-rsbx0));
        if ((ry0-rsby0)<0)and((rx0-rsbx0)>0)then psib:=arctan((ry0-
rsby0)/(rx0-rsbx0))+pi;
        axb:=ab*cos(psib);ayb:=ab*sin(psib);
        {suma cartesiana de las dos aceleraciones experimentadas}
        ax:=axa+axb;ay:=aya+ayb;
    end;
procedure calcula_psi2;
{ ngulos entre el planeta 2 y soles AyB y aceleraciones sufridas}
var psia,psib:real;
begin
    if ((r2y0-rsay0)>0)and((r2x0-rsax0)>0)then psia:=arctan((r2y0-
rsay0)/(r2x0-rsax0))+pi;
    if ((r2y0-rsay0)>0)and((r2x0-rsax0)<0)then psia:=arctan((r2y0-
rsay0)/(r2x0-rsax0));
    if ((r2y0-rsay0)<0)and((r2x0-rsax0)<0)then psia:=arctan((r2y0-
rsay0)/(r2x0-rsax0));
    if ((r2y0-rsay0)<0)and((r2x0-rsax0)>0)then psia:=arctan((r2y0-
rsay0)/(r2x0-rsax0))+pi;
    axa:=a2a*cos(psia);aya:=a2a*sin(psia);
    if ((r2y0-rsby0)>0)and((r2x0-rsbx0)>0)then psib:=arctan((r2y0-
rsby0)/(r2x0-rsbx0))+pi;
    if ((r2y0-rsby0)>0)and((r2x0-rsbx0)<0)then psib:=arctan((r2y0-
rsby0)/(r2x0-rsbx0));
    if ((r2y0-rsby0)<0)and((r2x0-rsbx0)<0)then psib:=arctan((r2y0-
rsby0)/(r2x0-rsbx0));
    if ((r2y0-rsby0)<0)and((r2x0-rsbx0)>0)then psib:=arctan((r2y0-
rsby0)/(r2x0-rsbx0))+pi;
    axb:=a2b*cos(psib);ayb:=a2b*sin(psib);
    a2x:=axa+axb;a2y:=aya+ayb;
end;
procedure dibuja_asteroide(x,y,c:integer);
begin
    PutPixel(MaxX div 2 +x, MaxY div 2 +y, c);
    PutPixel(MaxX div 2 +x+2, MaxY div 2 +y, c);
    PutPixel(MaxX div 2 +x, MaxY div 2 +y+2, c);
    PutPixel(MaxX div 2 +x-2, MaxY div 2 +y, c);
    PutPixel(MaxX div 2 +x, MaxY div 2 +y-2, c);
end;
procedure borra_asteroide(x,y,c:integer);
begin
    PutPixel(MaxX div 2 +x, MaxY div 2 +y, c);
    PutPixel(MaxX div 2 +x+2, MaxY div 2 +y, 0);
    PutPixel(MaxX div 2 +x, MaxY div 2 +y+2, 0);
    PutPixel(MaxX div 2 +x-2, MaxY div 2 +y, 0);
    PutPixel(MaxX div 2 +x, MaxY div 2 +y-2, 0);
end;

```

```

procedure calculos1;
var x1,x2,x3,y1,y2,y3,x4,y4:integer;
begin
    calcula_phy;
    SetColor(0);
    line(MaxX div 2 +round(r2x0*e*0.8), MaxY div 2 -round(r2y0*e), MaxX div 2
+round(arsax0*e*0.8), MaxY div 2 -round(arsay0*e));
    line(MaxX div 2 +round(r2x0*e*0.8), MaxY div 2 -round(r2y0*e), MaxX div 2
+round(arsbx0*e*0.8), MaxY div 2 -round(arsby0*e));
    {rsa0:=sqrt(rsax0*rsax0+rsay0*rsay0);
    asa:=g*mb/(rab*rab);psisa:=physb0;
    asax:=asa*cos(psisa);asay:=asa*sin(psisa);

    rsax0:=rsax0+vsax0*t+(asax*t*t)/2;rsay0:=rsay0+vsay0*t+(asay*t*t)/2;}
    arsax0:=rsax0;arsay0:=rsay0;
    physa0:=physa0+t*vsby0/4E+10;if physa0>2*pi then physa0:=physa0-
2*pi;
    rsax0:=4E+10*cos(physa0);rsay0:=4E+10*sin(physa0);
    dibuja_sol(round(arsax0*e*0.8),-round(arsay0*e),0);
    dibuja_sol(round(rsax0*e*0.8),-round(rsay0*e),14);
    {vsax0:=vsax0+asax*t;vsay0:=vsay0+asay*t;}

    {rsb0:=sqrt(rsbx0*rsbx0+rsby0*rsby0);
    asb:=g*ma/(rab*rab);psisb:=physa0;
    asbx:=asb*cos(psisb);asby:=asb*sin(psisb);

    rsbx0:=rsbx0+vsbx0*t+(asbx*t*t)/2;rsby0:=rsby0+vsby0*t+(asby*t*t)/2;}
    arsbx0:=rsbx0;arsby0:=rsby0;
    physb0:=physb0+t*vsby0/4E+10;if physb0>2*pi then physb0:=physb0-
2*pi;
    rsbx0:=4E+10*cos(physb0);rsby0:=4E+10*sin(physb0);
    dibuja_sol(round(arsbx0*e*0.8),-round(arsby0*e),0);
    dibuja_sol(round(rsbx0*e*0.8),-round(rsby0*e),14);
    {vsbx0:=vsbx0+asbx*t;vsby0:=vsby0+asby*t;}

{planeta2}
    borra_asteroide(round(r2x0*e*0.8),-round(r2y0*e), 4);
    r2pa0:=sqrt((r2x0-rsax0)*(r2x0-rsax0)+(r2y0-rsay0)*(r2y0-rsay0));
    r2pb0:=sqrt((r2x0-rsbx0)*(r2x0-rsbx0)+(r2y0-rsby0)*(r2y0-rsby0));
    a2a:=g*ma/(r2pa0*r2pa0);a2b:=g*mb/(r2pb0*r2pb0);
    calcula_psi2;
    r2x0:=r2x0+v2x0*t+(a2x*t*t)/2;r2y0:=r2y0+v2y0*t+(a2y*t*t)/2;
    x1:=round(r2x0*e*0.8);y1:=round(r2y0*e);
    x2:=round((r2x0*e+axa*20000)*0.8);y2:=round(r2y0*e+aya*20000);
    x3:=round((r2x0*e+axb*20000)*0.8);y3:=round(r2y0*e+ayb*20000);
    x4:=round((r2x0*e+a2x*20000)*0.8);y4:=round(r2y0*e+a2y*20000);
    SetColor(14);SetLineStyle(SolidLn,0,NormWidth);
    line(MaxX div 2+x1,MaxY div 2-y1,MaxX div 2+x2,MaxY div 2-y2);
    line(MaxX div 2+x1,MaxY div 2-y1,MaxX div 2+x3,MaxY div 2-y3);
    SetColor(3);
    line(MaxX div 2+x1,MaxY div 2-y1,MaxX div 2+x4,MaxY div 2-y4);
    SetLineStyle(DottedLn,0,NormWidth);SetColor(4);
    line(MaxX div 2 +round(r2x0*e*0.8), MaxY div 2 -round(r2y0*e), MaxX div 2
+round(arsax0*e*0.8), MaxY div 2 -round(arsay0*e));
    line(MaxX div 2 +round(r2x0*e*0.8), MaxY div 2 -round(r2y0*e), MaxX div 2
+round(arsbx0*e*0.8), MaxY div 2 -round(arsby0*e));
    dibuja_asteroide(round(r2x0*e*0.8),-round(r2y0*e), 12);

```

```

        PutPixel(MaxX div 2 +round(r2x0*e*0.8), MaxY div 2 -
round(r2y0*e), 15);
        v2x0:=v2x0+a2x*t;v2y0:=v2y0+a2y*t;

        SetColor(0);SetLineStyle(SolidLn,0,NormWidth);
        line(MaxX div 2+x1,MaxY div 2-y1,MaxX div 2+x2,MaxY div 2-y2);
        line(MaxX div 2+x1,MaxY div 2-y1,MaxX div 2+x3,MaxY div 2-y3);
        line(MaxX div 2+x1,MaxY div 2-y1,MaxX div 2+x4,MaxY div 2-y4);
    end;
procedure calculos(c:integer);
begin
    calcula_phy;
    {SetColor(0);
        line(MaxX div 2 +round(rx0*e), MaxY div 2 -round(ry0*e), MaxX div
2 +round(arsax0*e), MaxY div 2 -round(arsay0*e));
        line(MaxX div 2 +round(rx0*e), MaxY div 2 -round(ry0*e), MaxX div
2 +round(arsbx0*e), MaxY div 2 -round(arsby0*e));}

    arsax0:=rsax0;arsay0:=rsay0;
    physa0:=physa0+t*vsby0/4E+10;if physa0>2*pi then physa0:=physa0-
2*pi;
    rsax0:=4E+10*cos(physa0);rsay0:=4E+10*sin(physa0);
    dibuja_sol(round(arsax0*e*0.8),-round(arsay0*e),0);
    dibuja_sol(round(rsax0*e*0.8),-round(rsay0*e),14);

    arsbx0:=rsbx0;arsby0:=rsby0;
    physb0:=physb0+t*vsby0/4E+10;if physb0>2*pi then physb0:=physb0-
2*pi;
    rsbx0:=4E+10*cos(physb0);rsby0:=4E+10*sin(physb0);
    dibuja_sol(round(arsbx0*e*0.8),-round(arsby0*e),0);
    dibuja_sol(round(rsbx0*e*0.8),-round(rsby0*e),14);
    {Marte}
    {borra_asteroide(round(r2x0*e*0.8),-round(r2y0*e), 4);
    r2pa0:=sqrt((r2x0-rsax0)*(r2x0-rsax0)+(r2y0-rsay0)*(r2y0-rsay0));
    r2pb0:=sqrt((r2x0-rsbx0)*(r2x0-rsbx0)+(r2y0-rsby0)*(r2y0-rsby0));
    a2a:=g*ma/(r2pa0*r2pa0);a2b:=g*mb/(r2pb0*r2pb0);
    calcula_psi2;
    r2x0:=r2x0+v2x0*t+(a2x*t*t)/2;r2y0:=r2y0+v2y0*t+(a2y*t*t)/2;
    v2x0:=v2x0+a2x*t;v2y0:=v2y0+a2y*t;
    dibuja_asteroide(round(r2x0*e*0.8),-round(r2y0*e), 12);}
    {Tierra casi-inestable}
    rpa0:=sqrt((rx0-rsax0)*(rx0-rsax0)+(ry0-rsay0)*(ry0-rsay0));
    rpb0:=sqrt((rx0-rsbx0)*(rx0-rsbx0)+(ry0-rsby0)*(ry0-rsby0));
    aa:=g*ma/(rpa0*rpa0);ab:=g*mb/(rpb0*rpb0);
    calcula_psi;
    borra_asteroide(round(rx0*e*0.8),-round(ry0*e), c);
    rx0:=rx0+vx0*t+(ax*t*t)/2;ry0:=ry0+vy0*t+(ay*t*t)/2;
    PutPixel(MaxX div 2 +round(rx0*e*0.8), MaxY div 2 -round(ry0*e),
c);
    dibuja_asteroide(round(rx0*e*0.8),-round(ry0*e), c);
    vx0:=vx0+ax*t;vy0:=vy0+ay*t;
end;
procedure calcula_phy_soles;
begin
    if (rsay0>0)and(rsax0>0)then physa0:=arctan(rsay0/rsax0);    {sol
A}
    if (rsay0>0)and(rsax0<0)then physa0:=arctan(rsay0/rsax0)+pi;
    if (rsay0<0)and(rsax0<0)then physa0:=arctan(rsay0/rsax0)+pi;

```

```

        if (rsay0<0)and(rsax0>0)then physa0:=arctan(rsay0/rsax0);

        physb0:=physa0+pi; {sol
B}
        if physb0>2*pi then physb0:=physb0-2*pi; {lado opuesto al cd
masas}
        end;
procedure calculos0;
begin
    calcula_phy_soles;

    arsax0:=rsax0;arsay0:=rsay0;
    rab:=sqrt((rsax0-rsbx0)*(rsax0-rsbx0)+(rsay0-rsby0)*(rsay0-
rsby0));
    SetColor(0);
{line(MaxX div 2 +round(rsax0*e*0.8),MaxY div 2 -round(rsay0*e),MaxX div
2 +round(rsbx0*e*0.8),MaxY div 2 -round(rsby0*e));}
    borra_asteroide(round(arsax0*e*0.8),-round(arsay0*e),14);
    {PutPixel(MaxX div 2 +round(rsax0*e*0.8), MaxY div 2 -
round(rsay0*e), 14);}
    rsa0:=sqrt(rsax0*rsax0+rsay0*rsay0);
    asa:=g*mb/(rab*rab);psisa:=physb0;
    asax:=asa*cos(psisa);asay:=asa*sin(psisa);

rsax0:=rsax0+vsax0*t+(asax*t*t)/2;rsay0:=rsay0+vsay0*t+(asay*t*t)/2;
    dibuja_asteroide(round(rsax0*e*0.8),-round(rsay0*e),14);
    vsax0:=vsax0+asax*t;vsay0:=vsay0+asay*t;

    arsbx0:=rsbx0;arsby0:=rsby0;
    borra_asteroide(round(arsbx0*e*0.8),-round(arsby0*e),14);
    {PutPixel(MaxX div 2 +round(rsbx0*e*0.8), MaxY div 2 -
round(rsby0*e), 14);}
    rsb0:=sqrt(rsbx0*rsbx0+rsby0*rsby0);
    asb:=g*ma/(rab*rab);psisb:=physa0;
    asbx:=asb*cos(psisb);asby:=asb*sin(psisb);

rsbx0:=rsbx0+vsbx0*t+(asbx*t*t)/2;rsby0:=rsby0+vsby0*t+(asby*t*t)/2;
    dibuja_asteroide(round(rsbx0*e*0.8),-round(rsby0*e),14);
    vsbx0:=vsbx0+asbx*t;vsby0:=vsby0+asby*t;
    SetLineStyle(DashedLn,0,NormWidth);SetColor(8);
{line(MaxX div 2+round(rsax0*e*0.8),MaxY div 2-round(rsay0*e),MaxX div
2+round(rsbx0*e*0.8),MaxY div 2-round(rsby0*e));}
    gotoxy(50,18);Write('Distancia (A,B): ');
    gotoxy(49,19);Write(rab);
    gotoxy(50,20);Write('Va: ',sqrt(vsax0*vsax0+vsay0*vsay0));
    gotoxy(50,21);Write('Vb: ',sqrt(vsbx0*vsbx0+vsby0*vsby0));

    gotoxy(1,22);Write('Energia p: ',-g*ma/1E10*mb/1E10/rab);
    gotoxy(1,23);Write('Energia c:
',ma*(vsax0*vsax0/1E10+vsay0*vsay0/1E10));
    ep:=-
g*ma/1E10*mb/1E10/rab;ec:=ma*(vsax0*vsax0/1E10+vsay0*vsay0/1E10);
    gotoxy(1,24);write('Energia t: ',ep+ec);}
    end;
begin
    Inicia_modos_grafico;Presentacion;
    DatosIniciales;vsay0:=-27000;vsby0:=27000;rsax0:=-
4E+10;rsbx0:=4E+10;

```

```

        {vsay0:=-20000;vsby0:=20000;rsax0:=-
6E+10;rsbx0:=6E+10;
        vsay0:=-28804;vsby0:=28804;rsax0:=-
4E+10;rsbx0:=4E+10;
        vsay0:=-27000;vsby0:=27000;rsax0:=-
6E+10;rsbx0:=6E+10;}
        dibuja_los_ejes;t:=1000;
        repeat calculos0 until keypressed;
        readLn;ClearViewPort;
        DatosIniciales;vsay0:=-27000;vsby0:=27000;rsax0:=-
4E+10;rsbx0:=4E+10;
        vsay0:=-20000;vsby0:=20000;rsax0:=-
6E+10;rsbx0:=6E+10;
        {vsay0:=-28804;vsby0:=28804;rsax0:=-
4E+10;rsbx0:=4E+10;
        vsay0:=-27000;vsby0:=27000;rsax0:=-
6E+10;rsbx0:=6E+10;}
        dibuja_los_ejes;t:=1000;
        repeat calculos0 until keypressed;
        readLn;ClearViewPort;
        DatosIniciales;vsay0:=-27000;vsby0:=27000;rsax0:=-
4E+10;rsbx0:=4E+10;
        vsay0:=-20000;vsby0:=20000;rsax0:=-
6E+10;rsbx0:=6E+10;
        vsay0:=-28804;vsby0:=28804;rsax0:=-
4E+10;rsbx0:=4E+10;
        {vsay0:=-27000;vsby0:=27000;rsax0:=-
6E+10;rsbx0:=6E+10;}
        dibuja_los_ejes;t:=1000;
        repeat calculos0 until keypressed;
        readLn;ClearViewPort;
        DatosIniciales;vsay0:=-27000;vsby0:=27000;rsax0:=-
4E+10;rsbx0:=4E+10;
        vsay0:=-20000;vsby0:=20000;rsax0:=-
6E+10;rsbx0:=6E+10;
        vsay0:=-28804;vsby0:=28804;rsax0:=-
4E+10;rsbx0:=4E+10;
        vsay0:=-27000;vsby0:=27000;rsax0:=-
6E+10;rsbx0:=6E+10;
        dibuja_los_ejes;t:=1000;
        repeat calculos0 until keypressed;
        {Planeta estable Marte descomposicionn de fuerzas}
        readLn;ClearViewPort;DatosIniciales;
        dibuja_los_ejes;t:=2000;
        repeat calculos1 until keypressed;
        {Planeta terrestre inestable}
        ReadLn;ClearViewPort;DatosIniciales;
        dibuja_los_ejes;t:=1000;vx0:=-30951;
        repeat calculos(3) until keypressed;
        {Planeta cuasi estable la Tierra}
        ReadLn;ClearViewPort;DatosIniciales;
        dibuja_los_ejes;t:=1000;
        repeat calculos(3) until keypressed;
        {Planeta inestable Venus}
        ReadLn;ClearViewPort;DatosIniciales;
        rx0:=0;ry0:=1.08E+11;      {planeta0 amarillo: Venus}
        r0:=sqrt(rx0*rx0+ry0*ry0);
        vx0:=-52650;vy0:=0;      {-52600 -53000 cualquier rango inestable}

```



```
    dibuja_los_ejes;t:=1000;  
    repeat calculos(14) until keypressed;  
end.
```