HP Prime Calculator Programs

Dr. D.G. Simpson
Department of Physical Sciences and Engineering
Prince George's Community College

May 3, 2014

These calculator programs are written for the Hewlett-Packard Prime scientific calculator.

Contents

1 Projectile Problem

This program solves the projectile problem: given a target sitting on a hill at coordinates (x_t, y_t) and a cannon with muzzle velocity v_0, at what angle θ should the cannon be aimed to hit the target? The solution is found numerically using Newton's method. This is a very simple implementation—it includes no convergence test, and simply performs 15 iterations of Newton's method.

Enter the program below with the name `PROJTL`. Then run the program, and you will be prompted to enter v_0, x_t, y_t and θ_0, where v_0, x_t, and y_t are in any consistent set of units, and θ_0 (the initial estimate of the launch angle) is in degrees. The program returns the launch angle θ in degrees.

After running the program, the calculator will be set to degrees mode.

Program Listing

```
EXPORT PROJTL()
BEGIN
LOCAL V,X,Y,θ,J;
0▶HAngle;
INPUT(V);
INPUT(X);
INPUT(Y);
INPUT(θ);
θ*π/180▶ θ;
FOR J FROM 1 TO 15 DO
θ-(X*SIN(2*θ)-2*Y*(COS(θ))²-9.8*(X/V)²)/(2*X*COS(2*θ)+2*Y*SIN(2*θ))▶ θ;
END;
θ*180/π ▶ θ;
PRINT;
PRINT("θ="+θ);
1▶HAngle;
END;
```

Example. Let $v_0 = 30$ m/s, $(x_t, y_t) = (50$ m, 20 m$)$, and $\theta_0 = 30°$. Enter the above program with the name `PROJTL`. Run the program, and enter these values for `V`, `X`, `Y`, and θ. The program returns $\theta = 41.5357079292°$.

2 Kepler's Equation

Given the mean anomaly M (in degrees) and the orbit eccentricity e, this program solves Kepler's equation

$$M = E - e \sin E$$

to find the eccentric anomaly E. This is a very simple implementation—it includes no convergence test, and simply solves Kepler's equation by performing 15 iterations of Newton's method.

Enter the program with the name `KEPLER`. Then run the program, and you will be prompted to enter: M and e, where M is in degrees. The program returns the eccentric anomaly E in degrees.

After running the program, the calculator will be set to degrees mode.

Program Listing

```
EXPORT KEPLER()
BEGIN
LOCAL M,E,A,J;
0▶HAngle;
INPUT(M);
INPUT(E);
M*π/180▶M;
M▶A;
FOR J FROM 1 TO 15 DO
A-(M-A+E*SIN(A))/(E*COS(A)-1)▶A;
END;
A*180/π ▶A;
PRINT;
PRINT("EA="+A);
1▶HAngle;
END;
```

Example. Let $M = 60°$, $e = 0.15$. Enter the above program with the name `KEPLER`. Run the program, and enter these values for `M` and `E`. The program returns $E = 67.9666848988°$.

3 Hyperbolic Kepler's Equation

Given the mean anomaly M (in degrees) and the orbit eccentricity e, this program solves the hyperbolic Kepler equation

$$M = e \sinh F - F$$

to find the variable F. This is a very simple implementation—it includes no convergence test, and simply solves the hyperbolic Kepler equation by performing 15 iterations of Newton's method.

Enter the program with the name `HKEPLER`. Then run the program, and you will be prompted to enter: M and e, where M is in degrees. The program returns the variable F.

Program Listing

```
EXPORT HKEPLER()
BEGIN
LOCAL M,E,A,J;
INPUT(M);
INPUT(E);
M*π/180►M;
M►A;
FOR J FROM 1 TO 15 DO
A-(M-E*SINH(A)+A)/(1-E*COSH(A))►A;
END;
PRINT;
PRINT("F="+A);
END;
```

Example. Let $M = 60°$, $e = 1.15$. Enter the above program with the name `HKEPLER`. Run the program, and enter these values for `M` and `E`. The program returns $F = 1.55551859438$.

4 Barker's Equation

Given the constant $K = \sqrt{GM/(2q^3)}(t - T_p)$, this program solves Barker's equation

$$\tan\left(\frac{f}{2}\right) + \frac{1}{3}\tan^3\left(\frac{f}{2}\right) = \sqrt{\frac{GM}{2q^3}}\,(t - T_p)$$

to find the true anomaly f.

Enter the program with the name `BARKER`. Then run the program, and you will be prompted to enter the dimensionless number

$$K = \sqrt{\frac{GM}{2q^3}}\,(t - T_p)$$

followed by `ENTER`. The program returns the true anomaly f.

The program will work in either degrees or radians mode.

Program Listing

```
EXPORT BARKER()
BEGIN
LOCAL K;
INPUT(K);
1.5*ABS(K)▶A;
√(1+A*A)+A▶B;
3 NTHROOT B▶C;
(C*C-1)/(2*C)▶D;
2*D▶E;
2*ATAN(E)▶F;
PRINT;
PRINT("F="+F);
END;
```

Example. Let $K = 19.38$, and put the calculator in degrees mode. Enter the above program with the name `BARKER`. Run the program, and enter this value for `K`. The program returns $f = 149.084724939°$.

5 Reduction of an Angle

This program reduces a given angle to the range $[0, 360°)$ in degrees mode, or $[0, 2\pi)$ in radians mode. It will work correctly whether the calculator is set for degrees or radians mode.

Enter the program below with the name `REDUCE`. Then run the program, and you will be prompted to enter θ (in either degrees or radians) followed by `ENTER`. The program will return the equivalent reduced angle.

Program Listing

```
EXPORT REDUCE()
BEGIN
LOCAL θ,T,R;
INPUT(θ);
2*ACOS(-1)►T;
CASE
IF θ <0 THEN θ+(IP(-θ/T)+1)*T►R; END
IF θ ≥T THEN θ-IP(θ/T)*T►R; END
DEFAULT θ ►R;
END;
PRINT;
PRINT("θ="+R);
END;
```

Example. Let $\theta = 5000°$, and set the calculator for degrees mode. Enter the above program with the name `REDUCE`. Run the program, and at the prompt `θ=?` enter `5000 ENTER`. The program returns $\theta = 320°$.

6 Helmert's Equation

Given the latitude θ (in degrees) and the elevation H (in meters), this program uses Helmert's equation to find the acceleration due to gravity g.

Enter the program below with the name HELMERT. Then run the program, and you will be prompted to enter θ and H, where θ is in degrees and H is in meters. The program returns the acceleration due to gravity g in m/s^2.

After running the program, the calculator will be set to degrees mode.

Program Listing

```
EXPORT HELMERT()
BEGIN
LOCAL θ,H,G;
1▶HAngle;
INPUT(θ);
INPUT(H);
9.80616-.025928*COS(2*θ)+6.9E-5*(COS(2*θ))²-3.086E-6*H▶G;
PRINT;
PRINT("g="+G);
END;
```

Example. Let $\theta = 38.898°$, $H = 53$ m. Enter the above program with the name HELMERT. Run the program, and enter these values for θ and H. The program returns $g = 9.80051852685$ m/s^2.

7 Pendulum Period

Given the length L and amplitude θ of a simple plane pendulum, this program finds the period T, using the arithmetic-geometric mean method.

Enter the program below with the name PEND. Then run the program, and you will be prompted to enter m_1 (as M), m_2 (as N) , v_{1i} (as V), and v_{2i} (as W), in any consistent set of units. The program returns the post-collision velocities v_{1f} and v_{2f} in the same units.

Program Listing

```
EXPORT PEND()
BEGIN
LOCAL L,T,N,A,B,G;
1▶HAngle;
INPUT(L);
INPUT(θ);
0.5*(1+COS(0.5*θ))▶A;
√(COS(0.5*θ))▶G;
FOR N FROM 1 TO 10 DO
0.5*(A+G)▶B;
√(A*G)▶G;
B▶A;
END;
2*π*√(L/9.8)/A▶T;
PRINT;
PRINT("T="+T);
END;
```

Example. Let $L = 1.2$ m and $\theta = 65°$. Enter the above program with the name PEND. Run the program, and enter these values for L and θ. The program returns $T = 2.38976949659$ sec.

8 1D Perfectly Elastic Collisions

Given the masses m_1 and m_2 of two bodies and their initial velocities v_{1i} and v_{2i}, this program finds the post-collision velocities v_{1f} and v_{2f}, using

$$v_{1f} = \left(\frac{m_1 - m_2}{m_1 + m_2}\right) v_{1i} + \left(\frac{2m_2}{m_1 + m_2}\right) v_{2i}$$
$$v_{2f} = \left(\frac{2m_1}{m_1 + m_2}\right) v_{1i} + \left(\frac{m_2 - m_1}{m_1 + m_2}\right) v_{2i}$$

Enter the program below with the name ELAS1D. Then run the program, and you will be prompted to enter:

To run the program, execute program ELAS1D. At the prompts, enter the masses m_1 (as M) and m_2 (as N), and the initial velocities v_{1i} (as V) and v_{2i} (as W), in consistent units. The program will return the post-collision velocities v_{1f} and v_{2f} in the same units.

Program Listing

```
EXPORT ELAS1D()
BEGIN
LOCAL M,N,V,W,X,Y,Z;
INPUT(M);
INPUT(N);
INPUT(V);
INPUT(W);
M+N▶Z;
(M-N)/Z*V+2*N*W/Z▶X;
2*M*V/Z+(N-M)/Z*W▶Y;
PRINT;
PRINT("V1F="+X);
PRINT("V2F="+Y);
END;
```

Example. Let $m_1 = 2.0$ kg, $m_2 = 7.0$ kg, $v_{1i} = 4.0$ m/s, and $v_{2i} = -5.0$ m/s. Enter the above program with the name ELAS1D. At the prompts, enter: $m_1 =$ M $= 2.0$ kg; $m_2 =$ N $= 7.0$ kg; $v_{1i} =$ V $= 4.0$ m/s; and $v_{2i} =$ W $= -5.0$ m/s. The program returns $v_{1f} = -10$ m/s and $v_{2f} = -1$ m/s.