TI-89 / TI-92 / Voyage 200 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 Texas Instruments Voyage 200 graphing calculator and similar models with the Motorola 68000 processor (TI-89 and TI-92), using the built-in TI-BASIC language. Refer to the "Programming" chapter of the *Voyage 200 Graphing Calculator* manual for instructions on entering and running a program in the 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.

To run the program, execute program `projtl()`. At the prompt `v=?` enter the muzzle velocity `ENTER`. At the prompt `x=?` enter the horizontal coordinate of the target x_t followed by `ENTER`. At the prompt `y=?` enter the vertical coordinate of the target y_t followed by `ENTER`. At the prompt `θ=?` enter the initial estimate of the launch angle θ_0 followed by `ENTER`. The program returns the launch angle θ in degrees.

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

Program Listing

```
:projtl()
:Prgm
:setMode("Angle","Radian")
:Prompt v,x,y,θ
:θ*π/180→ θ
:For j,1,15
:  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
:EndFor
:θ*180/π → θ
:Disp "θ=",θ
:setMode("Angle","Degree")
:EndPrgm
```

Example. Let $v_0 = 30$ m/s, $(x_t, y_t) = (50 \text{ m}, 20 \text{ m})$, and $\theta_0 = 30°$. Enter the above program, and execute program `projtl()`. At the prompt `v=?` enter `30 ENTER`; at the prompt `x=?` enter `50 ENTER`; at the prompt `y=?` enter `20 ENTER`; and at the prompt `θ=?` enter `30 ENTER`. The program returns $\theta = 41.5357°$.

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.

To run the program, execute program `kepler()`. At the prompt `M=?` enter the mean anomaly M in degrees followed by `ENTER`. At the prompt `E=?` enter the eccentricity e followed by `ENTER`. The program returns the eccentric anomaly E in degrees.

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

Program Listing

```
:kepler()
:Prgm
:setMode("Angle","Radian")
:Prompt m,e
:m*π/180→m
:m→a
:For j,1,15
:  a-(m-a+e*sin(a))/(e*cos(a)-1)→a
:EndFor
:a*180/π →a
:Disp "EA=",a
:setMode("Angle","Degree")
:EndPrgm
```

Example. Let $M = 60°$, $e = 0.15$. Enter the above program, and execute program `kepler()`. At the prompt `M=?` enter `60 ENTER`; at the prompt `E=?` enter `.15 ENTER`. The program returns $E = 67.9667°$.

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.

To run the program, execute program `HKEPLER`. At the prompt `M=?` enter the mean anomaly M in degrees followed by `ENTER`. At the prompt `E=?` enter the eccentricity e followed by `ENTER`. The program returns the variable F.

Program Listing

```
:hkepler()
:Prgm
:Prompt m,e
:m*π/180→m
:m→a
:For j,1,15
:  a-(m-e*sinh(a)+a)/(1-e*cosh(a))→a
:EndFor
:Disp "F=",a
:EndPrgm
```

Example. Let $M = 60°$, $e = 1.15$. Enter the above program, and execute program `kepler()`. At the prompt `M=?` enter `60 ENTER`; at the prompt `E=?` enter `1.15 ENTER`. The program returns $F = 1.55552$.

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.

To run the program, execute program `barker()`. At the prompt `K=?` enter the dimensionless number

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

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

The program will work in either Degrees or Radians mode.

Program Listing

```
:barker()
:Prgm
:Prompt k
:1.5*abs(k)→a
:√(1+a*a)+a→b
:b^(1/3)→c
:(c*c-1)/(2*c)→d
:2*d→e
:If k<0 Then
:  −e→e
:EndIf
:2*tan⁻¹(e)→f
:Disp "F=",f
:EndPrgm
```

Example. Let $K = 19.38$, and put the calculator in Degrees mode. Enter the above program, and execute program `barker()`. At the prompt `k=?` enter `19.38 ENTER`. The program returns $f = 149.085°$.

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.

To run the program, execute program `reduce()`. At the prompt `θ=?` enter the angle θ (in either degrees or radians) followed by `ENTER`. The program will return the equivalent reduced angle.

Program Listing

```
:reduce()
:Prgm
:Prompt θ
:2*cos⁻¹(-1)→t
:If θ < 0 Then
:  iPart(−θ/t)+1→v
:  θ+v*t→r
:ElseIf θ ≥t Then
:  iPart(θ/t)→v
:  θ-v*t→r
:Else
:  θ →r
:EndIf
:Disp "θ=",r
:EndPrgm
```

Example. Let $\theta = 5000°$, and set the calculator for Degrees mode. Enter the above program, press `PRGM` and execute program `REDUCE`. 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, this program uses Helmert's equation to find the acceleration due to gravity g.

To run the program, execute program `helmert()`. At the prompt `θ=?` enter the latitude θ in degrees followed by `ENTER`. At the prompt `h=?` enter the elevation H followed by `ENTER`. 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

```
:helmert()
:Prgm
:setMode("Angle","Degree")
:Prompt θ,h
:9.80616-.025928*cos(2*θ)+6.9E-5*(cos(2*θ))^2-3.086E-6*h→g
:Disp "g=",g
:EndPrgm
```

Example. Let $\theta = 38.898°$, $H = 53$ m. Enter the above program and execute program `helmert()`. At the prompt `θ=?` enter `38.898 ENTER`; at the prompt `h=?` enter `53 ENTER`. The program returns $g = 9.80052$ 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.

To run the program, execute program `pend()`. At the prompt `l=?` enter the pendulum length L in meters followed by `ENTER`. At the prompt θ`=?` enter the pendulum amplitude θ in degrees followed by `ENTER`. The program returns the period T in seconds.

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

Program Listing

```
:pend()
:Prgm
:setMode("Angle","Degree")
:Prompt l,θ
:.5*(1+cos(.5*θ))→a
:√(cos(.5*θ))→g
:For n,1,10
:   .5*(a+g)→b
:   √(a*g)→g
:   b→a
:EndFor
:2*π*√(l/9.8)/a→t
:Disp "T=",t
:EndPrgm
```

Example. Let $L = 1.2$ m and $\theta = 65°$. Enter the above program and execute program `pend()`. At the prompt `l=?` enter `1.2 ENTER`. At the prompt θ= enter `65 ENTER`. The program returns $T = 2.38977$ 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

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

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

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

```
:elas1d()
:Prgm
:Prompt m,n,v,w
:(m-n)/(m+n)*v+2*n*w/(m+n)→x
:2*m*v/(m+n)+(n-m)/(m+n)*w→y
:Disp "v1f=",x
:Disp "v2f=",y
:EndPrgm
```

Example. Enter the above program and execute program `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.