[9.21] Programs for simple equation libraries

An equation library is a collection of equations together with a user interface to choose one of the equations, then solve for one of the equation variables. The HP-48 series has a substantial built-in library, and a similar library can be added to the HP-49G. The simple programs in this tip provide a simple but similar equation library for the TI-89/92+. These library programs do not include graphics, which are extremely helpful, and they cannot be used to solve multiple equations for a single variable.

Equation libraries are more useful for numeric solutions, not symbolic solutions. Once a symbolic solution is found, it is not necessary to find the same variable again. However, multiple numeric solutions may be required for equations in engineering and science applications. Therefore, the equation library programs use the built-in numeric solver. Using the built-in solver simplifies the programs, because the solver extracts the equation variables automatically. I also use the method of tip [7.10] to automatically start the numeric solver after the equation is chosen. The built-in solver uses the equation which is currently in the system variable *eqn*, and you can save any equation to that variable.

You provide the equation libraries. The library consists of a label (or name) for each equation, and the equation itself. It is easier to choose the equation by name, for example, *pendulum period*, than by choosing the equation. The library is a matrix with the equation names in the first column, and the equations in the second column.

If you have only a few equations, you can quickly choose a particular equation from one list of equation names.. But if you have dozens or hundreds of equations, it is more convenient to first choose a group of equations, then the desired equation from that group. To meet both requirements I show two equation library programs. *eqlib()* uses a single list of equations. *eqlibx()* uses groups of equations.

Equation library with one menu

The program for the single-menu equation library is:

```
eqlib()
Prgm
©Eqn library, one menu
©14novØ1/dburkett@infinet.com
©main\eqdef[n,2] holds labels & equations

local k

© Initialize pop-up menu index to zero, to allow use of [ESC] key to quit
Ø→k

© Extract and display equation labels as a pop-up menu
popup mat▸list(subMat(main\eqdef,1,1,rowDim(main\eqdef),1)),k

© Quit if [ESC] pressed: no equation chosen
if k=Ø:return

© Set the equation and start the numeric solver
main\eqdef[k,2]→eqn
setMode("Split 1 App","Numeric Solver")

EndPrgm
```

This is a very simple equation library, but it can be implemented in only five functional lines of TI Basic.

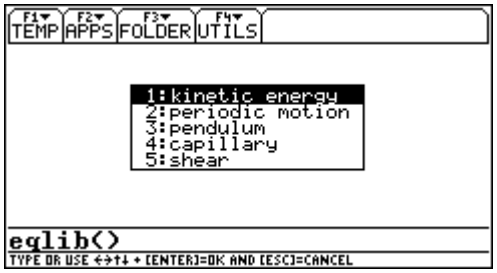
Before you can use *eqlib()*, you must set up the equation library in matrix *main\eqdef*. You can create this matrix in the entry line, or you can use the built-in matrix editor by pressing [APPS] [6] [3] to create a new matrix. Refer to the complete TI-89/92+ user's guide to learn how to use the matrix editor. As an example, *eqdef* might look like this:

	c1	c2
1	"kinetic energy"	.5*m*v^2=e
2	"periodic motion"	x=a*cos(ω*t+ϕ)
3	"pendulum"	t=2*π*√(l/(d*g*m))
4	"capillary"	h=2*cos(α)*τ/(g*ρ*r)
5	"shear"	f*tan(θ)/(a*g)=d

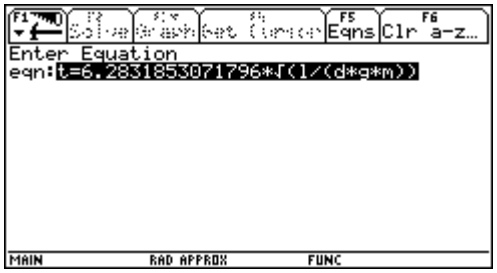
The first column *c1* holds the equation labels. These are the names that are shown in the pop-up menu to choose the equation, so use names such that you can remember the equation to which they apply. The names are strings with double-quote characters. The second column *c2* holds the equations themselves.

If you later need to edit your equation library, it is easily done in the matrix editor. You can use the [F6] Util menu to insert and delete rows.

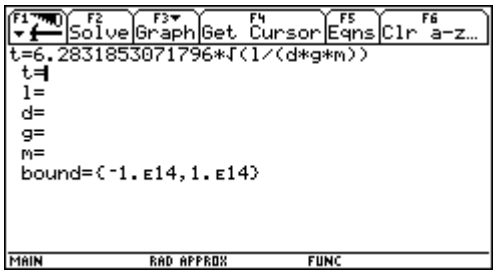
When you start *eqlib()*, this pop-up menu is shown:



You can press [ESC] at this point to exit the equation library and return to the home screen. If you press [3] to choose the pendulum equation, the numeric solver is started:



Press [ENTER] to accept this equation, and the numeric solver continues:



Note that the equation variables are displayed. Refer to the complete user's guide to learn how to use the numeric solver. In general, use [UP] and [DOWN] to choose the known variables and enter their values, then select the unknown variable and press [F2] Solve, to find the solution. The numeric solver works more quickly if you edit the *bound* variable list, and supply an initial guess for the unknown variable. For example, if you know that the solution is positive and less than 1000, you might set *bound* to {0,1000}. If you know that the solution should be about 120, then you would set the unknown variable to 120 before pressing [F2] to solve for it.

Unfortunately *bound* cannot be set from TI Basic. If this was possible, we could further automate the solution process by including a default *bound* setting in the equation library definition matrix.

Equation library with multiple menus

If your often use many equations, it is awkward to choose the correct equation from the single pop-up menu used in *eqlib()*. One solution is to use multiple equation library matrices to organize groups of equations. *eqlibx()*, shown below, implements this idea.

```
eqlibx()
Prgm
©Eqn library, multiple menus
©14novØ1/dburkett@infinet.com
©main\eqlxmenu{} holds menu names
©main\eqdef{m}[n,2] holds labels & equations; {m} = 1,2,3,...

local k,n

Ø→k                                              © Display 'menu of menus'
popup main\eqlxmenu,k
if k=Ø:return                                    © Quit if [ESC] pressed

"main\eqdef"&string(exact(k))→n                  © Build name of library matrix

Ø→k                                              © Display equation menu
popup mat▸list(subMat(#n,1,1,rowDim(#n),1)),k
if k=Ø:return                                    © Quit if [ESC] pressed

#n[k,2]→eqn                                      © Set chosen equation
setMode("Split 1 App","Numeric Solver")          © Start numeric solver

EndPrgm
```

This program displays two pop-up menus. The first menu shows the names of the different libraries. The second menu shows the equations in the chosen library. The library names are stored in the list variable *main\eqlxmenu*. For example, you might have three groups of equations for geometry, electricity, and physics, and you could use this for *main\eqlxmenu*:

```
{"geometry","electricity","physics"}
```

There is a corresponding equation matrix for each element of *eqlxmenu*, and the matrices have specific names. The library for the first element is *main\eqdef1*, for the second element is *main\eqdef2*, and so on. For our example, the equation matrices are

geometry: *main\eqdef1*
electricity: *main\eqdef2*
physics: *main\eqdef3*

The format of these equation matrices is the same as that of *eqlib()* above: the first column *c1* is the equation label, and the second column *c2* is the equation. These examples show typical equation matrices.

Matrix main\eqdef1 for geometry equations

	c1	c2
1	"cosines law"	c^2=a^2-2*a*b*cos(cc)+b^2
2	"triangle area"	.5*a*b*sin(cc)=area
3	"polygon area"	tan(π/k)*k*r^2=area

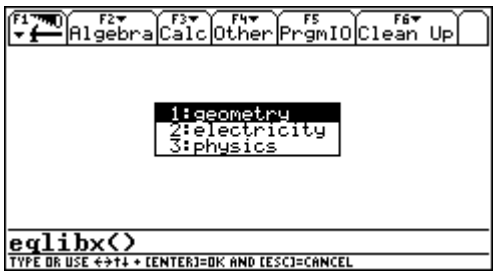
Matrix main\eqdef2 for electricity equations

	c1	c2
1	"Ohm's law"	v=i*r
2	"voltage divider"	vo=v*(rr2/(rr1+rr2))
3	"parallel resistance"	r=(rr1*rr2)/(rr1+rr2)

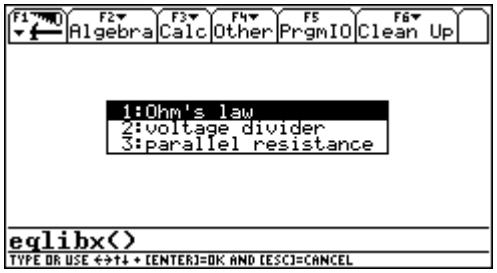
Matrix main\eqdef3 for physics equations

	c1	c2
1	"kinetic energy"	.5*m*v^2=e
2	"periodic motion"	x=a*cos(ω*t+ϕ)
3	"pendulum"	t=2*π*√(l/(d*g*m))

When *eqlibx()* is executed, the first menu is shown:



If we press [2] to choose the 'electricity' menu item, a second menu is shown, from which the desired equation is selected:



After the equation is selected, the numeric solver is started, and you can set and solve for the variables as needed.