As a further improvement to the program *(if you are interested in conserving steps in memory)*, rearrange the labels as shown below:

Keys	Comments	Keys	Comments
LBL		LBL	Beginning of common section.
A	This program calculates sin x and then branches to label 1.	1	
f		ENTER↑	
SIN		ENTER↑	
GTO		×	$\dfrac{a^2}{3a^2 + 2}$
1		3	
		×	
LBL	This program calculates cos x and then continues execution through label 1.	2	
B		+	
f		÷	
COS		RTN	End of both programs.

First of all, notice that LBL C has been replaced by LBL 1 . Since we are not planning on executing that portion of the program from the keyboard, it is not necessary to use a valuable program control key. Secondly, notice that we've eliminated two steps in the program by positioning LBL B directly before LBL 1 (*previously* LBL C). In this way, program execution doesn't have to transfer from LBL B to LBL 1 using GTO 1 , it can continue sequentially.

Subroutine Branching

A second method of transferring program execution is by means of subordinate programs or "subroutines." When a series of steps is repeated in a program or is common to a number of programs, a single subroutine containing the steps may be written.

Just as you use the A thru E keys to control the steps between the corresponding LBL and RTN , so can the calculator use these keys. When the RTN is reached, instead of stopping the program, program execution automatically branches back (*returns*) to the step following the original branch instruction.

Program A   Subroutine B
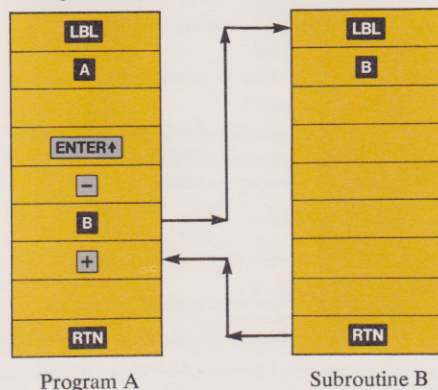
As you can see, a subroutine is a program. The only difference is the usage. In the above illustration, if you press B , the program controlled by B is executed and the calculator stops at the RTN . However, if you press A , the calculator executes the program controlled by A sequentially until it reaches the B program step. Then program execution transfers to LBL B . When the calculator reaches that same RTN this time, it now branches back to the A program and continues execution sequentially, starting with the step that follows the B key.

In other words, in the A program the B key is just one more key in the program. The program executes just as if the keys