Appendix A

In Case of Difficulty

In the event that you have difficulty with your calculator the following instructions will help you to analyze the problem. You may be able to correct your calculator problem without returning it to a service facility. If the suggested remedies are not successful, contact the Consumer Relations Department by mail or telephone (refer to the If You Have Questions or Need Assistance section later in this appendix). Please describe in detail the symptoms of your calculator.

If one of the following symptoms appears while operating with the optional printing unit, disconnect the printer. If the symptom disappears when the calculator is removed from the printing unit, refer to the printing unit manual.

Symptom	Remedy
1. Display is blank for no obvious reason and pressing [ON] has no effect.	Press and hold [R/S] momentarily. If display returns, the calculator was running a long program or hung in a loop.
2. Display shows erroneous results, stray segments, erratic numbers, grows dim, or goes blank	The batteries are probably discharged. Refer to the Battery Replacement section of this appendix.

3. Error appears while performing keyboard operations.	An invalid operation or key sequence has been entered or the limits of the calculator have been exceeded. See Appendix B for a list of these conditions.
4. The calculator produces incorrect results or Error when running a program.	The partition has been moved so that not all of the instructions are in program steps. Repartition for few enough data memories so that the program area is large enough. An illegal operation, overflow or underflow occurred while the program was running. Appendices B, C and D may be useful in finding the problem.
5. Calculator will NOT go into learn mode, single step, or list.	Calculator is partitioned for no program steps. Set the partitioning for less than 64 user data memories. Partitioning converted the current program step to part of a data memory and the program pointer is stranded. Press [RST] to recover the program pointer.

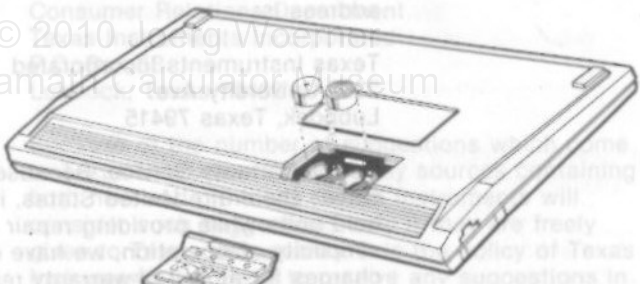
If the difficulty occurred while the printer was connected, the problem might be with the printer. Disconnect the printer, remove the printer's batteries, wait for a few minutes, reinstall the batteries, and reconnect the printer.

Battery Replacement

NOTE: The calculator cannot retain data in its user data memories or program memory when the batteries are removed or become discharged.

The calculator uses 2 of any of the following batteries for up to 750 hours of operation: Panasonic LR-44, Ray-O-Vac RW-82, Union Carbide (Eveready) A-76, or the equivalent. For up to 2000 hours of operation use Mallory 10L14 or D357, Union Carbide (Eveready) 357, Panasonic WL-14, Toshiba G-13, Ray-O-Vac RW-42, or the equivalent.

1. Turn the calculator off. Place a small screwdriver, paper clip, or other similar instrument into the slot and gently lift the battery cover.



2. Remove the discharged batteries and install new ones as shown.

3. Replace the cover top edge first, then gently press until the bottom of the cover snaps into place.

4. Press [ON], [CLR], [2nd] [CP], [2nd] [CMs], [2nd] [CSR], [2nd] [Part] 32, and [CLR]. The display then shows 0 and the calculator is ready to be used.

CAUTION: Do not incinerate the old batteries.

If the Difficulty Persists

Before returning your calculator for repair, try removing the batteries, waiting a few minutes, reinstalling the batteries, and seeing if the difficulty persists. For your protection, the calculator must be sent insured; Texas Instruments cannot assume any responsiblity for loss of or damage to uninsured shipments.

Please include information on the difficulty experienced with the calculator as well as return address information including name, address, city, state and zip code. The shipment should be carefully packaged, adequately protected against shock and rough handling and sent to one of the Texas Instruments Service Facilities listed with the warranty.

NOTE: The P.O. box number listed for the Lubbock Service Facility is for United States parcel post shipments only. If you use another carrier, the street address is:

Texas Instruments Incorporated
2305 University Ave.
Lubbock, Texas 79415

Out-of-Warranty Service. Because our Service Facility serves the entire United States, it is not feasible to hold units while providing repair estimates. For simplicity of operation, we have established flat-rate charges for all out-of-warranty repairs. To obtain the correct charges for a particular model, call our toll-free number listed in this section.

Exchange Centers

If your calculator requires service and you do not wish to return the unit to a service facility for repair or replacement, you may elect to exchange the unit for a factory-reconditioned calculator of the same model (or equivalent model specified by TI) by going in person to one of the exchange centers which have been established across the United States. A handling fee will be charged by the exchange center for in-warranty exchanges of the calculator. Out-of-warranty exchanges will be charged at the rates in effect at the time of the exchange.

To determine if there is an exchange center in your locality, look for Texas Instruments Incorporated Exchange Center in the white pages of your telephone directory or look under one of the following two headings in the yellow pages: "Calculating & Adding Machines & Supplies" or "Computers—Service & Repair." Please call the exchange center to check for the availability of your model, or write the Consumer Relations Department for further details and the location of the nearest exchange center.

If You Have Questions or Need Assistance
For technical questions such as programming, specific applications, etc., you can call (806) 747-3841. Please note that this is not a toll-free number and collect calls cannot be accepted.

As an alternative you can write to

Consumer Relations Department
Texas Instruments Incorporated
P.O. Box 53
Lubbock, Texas 79408

Because of the number of suggestions which come to Texas Instruments from many sources containing both new and old ideas, Texas Instruments will consider such suggestions only if they are freely given to Texas Instruments. It is the policy of Texas Instruments to refuse to receive any suggestions in confidence. Therefore, if you wish to share your suggestions with Texas Instruments or if you wish us to review any program which you have developed, please include the following statement in your letter.

"All of the information forwarded herewith is presented to Texas Instruments on a nonconfidential, nonobligatory basis; no relationship, confidential or otherwise, expressed or implied, is established with Texas Instruments by this presentation. Texas Instruments may use, copyright, distribute, publish, reproduce, or dispose of the information in any way without compensation to me."

Appendix B

Error Conditions

The display shows Error when overflow or underflow occurs or when an improper operation is requested. Proceeding without clearing the error leaves the display in a state where all decimals are turned on except the true decimal position. Pressing [CE] clears the error condition only. Pressing [CLR] clears the error condition, the display, and all pending operations. You must then determine what caused the error and re-key the entry to avoid the problem.

The following lists give the circumstances which cause "Error" to be displayed.

The first section shows the general conditions. The second section lists errors due to statistical operations.

Section 1— General Error Conditions

1. Number entry or calculation result (including in memories) is outside the range $-9.999999499999 \times 10^{99}$ to -1×10^{-99}, zero, or 1×10^{-99} to $9.999999499999 \times 10^{99}$.
2. Dividing a number by zero.
3. Calculating [log], [lnx], or [1/x] of 0 or calculating the zeroth root of any number (except zero).
4. Square root of a negative number.
5. Calculating [log], [lnx], a power or root of a negative number.
6. Tangent of 90° or 270°, π/2 radians or 3π/2 radians, 100 grads or 300 grads, or their rotational multiples such as 450°.
7. Using an argument outside the the following limits.

Function	Limit
arcsin x, arccos x	$-1 \leq x \leq 1$
lnx, log x	$1 \times 10^{-99} \leq x \leq 9.999999499999 \times 10^{99}$
e^x	$-227.9559242 \leq x \leq 230.2585092$
10^x	$-99 \leq x \leq 99.99999998$

8. Attempting to have more than 9 open levels of parentheses or more than 8 pending operations.
9. Calculating rectangular to polar conversions with values for X and Y such that the sum of their squares exceeds the upper or lower limit of the calculator.

10. Attempting to partition for more than 64 user data memories.
11. Attempting to access a memory that is not defined by the current partition.
12. Attempting to transfer to unassigned label positions or outside of the program partition to nonexistent program locations.
13. Calling for an OP code outside the range 00-08, 10-39.
14. Attempting to plot (OP 07) outside the range $0 \leq INT(X) \leq 15$.
15. Calling another subroutine when the return register is full.

Section 2— Statistical Error Conditions

1. Calculating standard deviation ([INV] [2nd] [$\overline{x}$]) with only one data point.
2. Entering a data point that has a value for x or y outside the range $x < \pm 1 \times 10^{-50}$ or $x > \pm 1 \times 10^{50}$.
3. Entering a series of data points such that the sum of their squares exceeds the upper or lower limit of the calculator.
4. Making it so that there are fewer than zero data points by removing data points with [INV] [2nd] [Σ+].
5. Calculating the slope, intercept, correlation, x′, or y′ of a line that parallels the y-axis (vertical line).
6. Calculating the correlation, x′, or y′ of a line that parallels the x-axis (horizontal line).
7. Calculating the slope, intercept, correlation, x′, or y′, with only one data point entered.

Errors Encountered When Running A Program

When any of the foregoing errors occur in a program, what happens next depends upon the programmer. Program halts are not an automatic consequence of an error condition except for number 10, 11, 12, 13, 14, or 15 in the first list. For halting errors, the number in the display when the error occurred is displayed. For errors that let execution continue, zero replaces any value in the display when the error occurred and is then used for subsequent calculations. If the programmer desires he may instruct the calculator to cease execution when an error conditon arises by setting flag 8 or by using the error tests, OP 18 and 19.

Appendix C

Accuracy Information

Each calculation produces a 13-digit result which is rounded to a 10-digit standard display. The 5/4 rounding technique used adds 1 to the least significant digit in the display if the next non-displayed digit is five or more. If this digit is less than five, no rounding occurs. In the absence of these extra digits, inaccurate results would frequently be displayed, such as

$$1 \div 3 \times 3 = 0.999999999$$

Because of rounding, the answer is given as 1, but is internally equal to 0.999999999999.

The higher order mathematical functions use iterative calculations. The cumulative error from these calculations in most cases is maintained beyond the 10-digit display so that no inaccuracy is displayed. Most calculations are accurate to ± 1 in the last displayed digit. There are a few instances in the solution of high order functions where display accuracy begins to deteriorate as the function approaches a discontinuous or undefined point. For example, the tangent of 87° is accurate for all displayed digits. However, the tangent of 89.9999996° is accurate to only six places. Another example is when the Y^x function has a Y value that approaches 1 and an X value that is a very large positive or negative number. The displayed result for 1.00000056^{400} is accurate for all displayed digits, while 1.00000056^{160000} is accurate to only eight places. In rectangular to polar conversions in any regular mode, values more than five orders of magnitude apart may be displayed having an angle of 0° or 90°.

Trigonometric values can be calculated for angles greater than one revolution. As long as the trigonometric function result is displayed in normal form rather than in scientific or engineering notation, all displayed digits are accurate for any angle from −36,000° to 36,000° and −40,000 to 40,000 grads. The equivalent range in radians (±200π) is comparable to degrees and grads in accuracy except at rotation multiples of π and π/2.

The rounded value of π limits accuracy at these points. In general, the accuracy decreases one digit for each decade outside this range.

Since periodic functions have the same magnitude at points one period apart, the inverses of these functions can have any number of correct answers. The inverse trigonometric functions of the calculator have predefined ranges given here.

Function	Range of Resultant Angle
arcsin of	
positive argument	0 to 90°, π/2 radians, or 100 grads
negative argument	0 to −90°, −π/2 radians, or −100 grads
arccos of	
positive argument	0 to 90°, π/2 radians, or 100 grads
negative argument	90° to 180°, π/2 to π radians, or 100 to 200 grads
arctan of	
positive argument	0 to 90°, π/2 radians, or 100 grads
negative argument	0 to −90°, −π/2 radians, or −100 grads

Appendix D

Troubleshooting Programs

First check to see that the program has been keyed in correctly. If the intended instructions have been keyed in correctly, the problem lies in the design of your program. A list of common programming errors is given in this appendix. Once you have studied these suggestions, you may recognize your mistake. If not, continue on into Program Diagnosis, the next part of this section.

The optional PC-200 printer becomes especially useful here for listing your program or tracing calculations. Flag 9 can be set either manually or in a program or [2nd] [Trace] can be pressed to trace all program calculations. This helps detect the source of program errors.

Basic Considerations

Algebraic Operating System
The algebraic hierarchy of the calculator causes operations to be completed in order of precedence. The expression $2 + 3 \times 6$ is interpreted as $2 + (3 \times 6) = 20$, not $(2 + 3) \times 6 = 30$.

Single-variable functions must follow the number being operated on. For example, sin π is evaluated by the sequence [2nd] [π] [2nd] [sin].

Equals Instruction—[=]
The equals instruction completes all pending operations; hence it should be used with discretion, especially in subroutines.

The lack of an equals can also cause problems. Consider the expression $(2 \times 3)^2$. The sequence [2] [×] [3] [x²] causes the calculator to determine the value 2×3^2. The sequence required here is either [2] [×] [3] [=] [x²] or [( ] [2] [×] [3] [ )] [x²].

Multiple Labels
Each label may be used only once within a program. The label search mechanism always begins at location 000, not at the point where the label is called. Therefore, transfer always goes to the first label and a second use of that label is never found.

Reset Instruction—[RST]
This instruction is very useful, but keep in mind all of the things it can do so you can avoid unintentional effects. [RST] performs three functions: positions the program pointer to location ST, resets all program flags, and clears the subroutine return register.

Statistical Functions
When using the preprogrammed statistical functions, data are summed into the contents of data memories 01—06. Therefore, a program using these functions should not only avoid the use of these memories, but should clear them as well before starting data entry. Steps should also be taken to preserve the value stored in the t register if it is needed later in the program.

Polar/Rectangular Conversions
Here the primary thing to remember is to select the correct angle mode.

Angle Mode Selection
Your calculator powers-up in the degree mode. If you want angles to be interpreted in either radians or grads you need to instruct the calculator to do so with the appropriate keys. The calculator then remains in the selected angle mode until another is chosen.

Functions Operating on the Display Only
[EE], [Fix], [2nd] [DMS-DD], and [INV] [2nd] [DMS-DD] operate only upon the contents of the display, not the display register. That is, any guard digits and any digits suppressed by placing the calculator in a fix-decimal display format are lost when you use these instructions.

T Register Comparisons
The display and t register are compared according to the current display format. The comparison of precise numbers might not give the expected result if the display format fixes the number of digits to fewer than needed.

Editing
You should use care when editing programs. Changes which may seem almost insignificant may cause complications elsewhere. Consider all possible effects that any change could have. Some things to watch out for are merged addresses, duplicate labels, and merged instructions (like [INV] [SBR]).

Adding and deleting instructions invariably moves parts of the program up and down. Transfer instructions using absolute addressing should be corrected accordingly. Remember that any time a step is keyed in, it is inserted moving all steps after that point in the program down one step.

Partitioning
Be sure that the data memory and program area you use are within your partition. A data memory equates to 8 program instructions. Be careful when repartitioning that the contents of a data memory do not become 8 program instructions and vice versa.

Program Diagnosis

The previous section is a list of common programming oversights. Now we shall discuss more elusive problems that might have occurred in your program.

Program Does Not Terminate
When a program does not terminate, it is said to be "hung" in a loop. One means of finding the infinite loop is to step through the program analyzing each instruction and give special attention to transfer statements. Though a branching instruction is likely to be the culprit, unconditional transfers should be checked first since an error here is easier to detect. Be wary of sequences such as [LBL] [D] . . . [GTO] [D]. Here a conditional transfer is required to provide a way out of the loop. Examine conditional transfers next, especially if they are designed to begin or end a loop.

The [2nd] [Dsz] instruction shouldn't send a program into an infinite loop unless the program does something to alter the contents of the data memory being decremented or the stored value is greater than 10^{12}. Make sure that you allow this data memory to go to zero. If a very large number is stored in the data memory that is being decremented, the program may take an exceptionally long time to terminate and only appear to be hung in a loop. However, if the program calculates the number of times to loop, you may need to examine the method used in determining this number.

Conditional transfers making t register comparisons should also be carefully examined. When you use a transfer of this type to control a loop, you expect your calculations to converge within predetermined limits. Naturally, if your calculations don't converge, the loop doesn't terminate. Check the equations and the instructions you are using to program these equations. Also, make sure that the correct test value is stored in the t register. If the program is designed to determine this number, you should examine these calculations as well. One additional problem (discussed previously in t register Comparisons under CONDITIONAL TRANSFERS in Chapter 4) is that these instructions compare only the display format so FIX or EE can make two values equal when compared even though internally they are slightly different.

If you are unable to detect any error after completing this evaluation, refer to "Using the Calculator in Diagnosis" later in this section.

Consistent Data Yields Inconsistent Results
Errors of this type are usually caused by conditional transfers that, when used improperly, can give correct results one time and not the next. As an illustration, consider the following program excerpts:

```
LBL
A
CP
X≥T
B
ST.F  Set flag 3 when X is negative
03
LBL
B
.
.
.
IF.F
03
C     Transfer to label C when flag 3 is set
.
.
.
LBL
C
.
.
```

Here, the programmer wants to skip a portion of the program if the entered number is negative. Let's assume that the following numbers are entered into the program: 12, −16, 12. The program produces correct results when the first two entries are made; however, the second time 12 is entered, execution does not follow the path for positive entries. The problem is that the program includes no provision for resetting flag 3 when positive entries are made. This situation may be remedied by placing the sequence INV ST.F 3 following LBL A.

Similar problems may occur with any transfer operation. Unfortunately, the problem is often unique to the situation. As a general rule, however, the first thing to do when diagnosing a problem of this nature is to look for a pattern in the answers.

Troubleshooting Programs
Of course, not all errors of this type are caused by transfer operations. Consider a situation in which consistently different answers (e.g., increasing according to a recognizable pattern) result from entering the same data several times in succession.

```
LBL
A
.
.
.
SBR
SUM
.
.
LBL
SUM
SUM
12
.
.
.
RTN
```

Here, the problem is caused by the careless use of a data memory. If memory 12 is never given a known starting value with a STO instruction or by clearing, each call of the subroutine accumulates a new total.

In most cases, such situations may be avoided by incorporating an initialization sequence into each of your programs. A thorough initialization procedure is shown below. Initialize by pressing [RST] [E] to first reset all flags and to then call the clearing subroutine.

Instruction	Comments
LBL	
E	
CMS	Clears data memories
CLR	Clears display
CP	Zeros t-register
INV	
FIX	Removes fix-decimal
RTN	Ends routine

Consistently Wrong Answers
It is possible that a program in which the same erroneous answers occur consistently, regardless of what data you enter, has been written using an incorrect solution. However, if you have manually worked through your equations, found them valid for all cases, and can find no instruction error, then refer to the following section.

Using the Calculator in Diagnosis
Once you have determined what values should be computed and displayed, and where they should be stored at different stages of the program, your calculator can be the means of examining a malfunctioning program.

A strong word of caution: When you recall data memory contents for examination at an inspection point, be sure that you restore the contents of the display register before continuing program execution. Otherwise, errors may occur which are not caused by the program. It is a good idea to use [2nd] [Exc] to call the data to the display for examination and then replace it, using the same sequence. This returns the most recently calculated value to the display.

There are several instructions that you may use to analyze a program as it is running. Inserting [R/S] commands at various key points in a program is a quick method of finding where an error first appears. When the program stops, you should check the display, memories, and t register and decide if they are correct.

When inserting R/S instructions, it is easiest to first insert at the last point in the program that you wish execution to stop, proceeding backwards. Inserting the last R/S first shifts only the program locations after the instruction. The program ahead of the insert remains in agreement with any listing you may be working from. Then it is easier to find the next position to put a R/S. Be careful that insertions do not invalidate any absolute addresses by shifting instructions from their expected positions.

Once a discrepancy is found, run the program and stop at the [R/S] instruction ahead of the [R/S] where the error was detected. Now use the [SST] key to execute the program one step at a time until you find the exact location of the error. After the error is corrected, delete the [R/S] instructions in the order they occur in program memory. Repeat this process until all errors have been corrected.

The most easily diagnosed problem is one that results in an error condition. When this occurs, simply set flag 8 and run the program again. When the error is encountered, the program halts. Pressing [LRN] shows you the program location where the error occurred. (Processing actually stops with the program pointer on the first location following the error.) You should then be able to determine the nature of the error and make necessary correction.

Using the Printer in Diagnosis

The optional printer is a valuable aid in diagnosing a program. With your calculator connected to the printer, here's what you can do.

1. Press [RST] [2nd] [List] to obtain a complete listing of your program instructions including program location numbers and instruction mnemonics. This saves you the trouble of single-stepping through a program in the learn mode to verify that you've entered your program correctly.

2. Run a program while in the trace mode. This enables you to easily follow the sequence of calculations on a step-by-step printout and discover exactly where the program deviates from the solution that you intended.

3. Press XX [INV] [2nd] [List] to obtain a list of data memory contents beginning with memory XX. By stopping a program at various key points and performing this operation, you can easily verify that the data memories contain the right quantities at the right times.

4. Press [RST] [OP] 08 and the printer lists all labels and their absolute addresses. By using this feature you won't have to search through entire program listings to find where your labels are located.

Appendix E

Instructions of a program appear as the following mnemonics. Remember to press [2nd] before using a second function.

Mnemonic (†)	Key	Meaning
†Mnemonic if different when printed		
(	[( ]	open parenthesis
)	[ )]	close parenthesis
+	[+]	plus
−	[−]	minus
×	[×]	times
/ (÷)	[÷]	divided by
=	[=]	equals
+/−	[+/−]	change sign
.	[.]	decimal
0	[0]	zero
1	[1]	one
2	[2]	two
3	[3]	three
4	[4]	four
5	[5]	five
6	[6]	six
7	[7]	seven
8	[8]	eight
9	[9]	nine
A	[A]	A—specifies a label
A′	[A′]	A′—specifies a label
ADV	[Adv]	Advance printer paper
B	[B]	B—specifies a label
B′	[B′]	B′—specifies a label
C	[C]	C—specifies a label
C′	[C′]	C′—specifies a label

CE		[CE]	clear entry
CLR		[CLR]	clear display and pending operations
CMS		[CMs]	clear memories
COS		[cos]	cosine
CP		[CP]	clear program
CSR		[CSR]	clear statistics registers
D		[D]	D—specifies a label
D'		[D']	D'—specifies a label
DEG		[Deg]	select degree units for angles
D.MS	(DMS-D)	[DMS-DD]	converts degrees.minutes seconds to decimal degrees
DSZ		[Dsz]	decrement and skip on zero (The decrement memory or the destination or both may be addressed indirectly.)
E		[E]	E—specifies a label
E'		[E']	E'—specifies a label
EE		[EE]	enter exponent of ten for scientific notation
ENG		[Eng]	select engineering notation
EXC		[Exc]	exchange display with memory
EX*	(EXC*)	[Exc] [Ind]	exchange indirect
Σ+		[Σ+]	accumulate data point for statistics
FIX		[Fix]	fixed decimal (may be specified indirectly)
GTO		[GTO]	go to label or step number
GO*	(GTO*)	[GTO] [Ind]	go to indirect
GRD	(GRAD)	[Grad]	select grad units for angles

IF.F	(IFF)	[IIF]	if flag is set (The flag or the destination or both may be addressed indirectly.)
IND		[Ind]	indirect addressing of operation
INT	(INTG)	[Intg]	retain only the integer portion of the displayed number
INV		[INV]	inverse function
LBL		[LBL]	label program step: the next step contains the label name
LNX	(LnX)	[lnx]	natural log, base e
LOG		[log]	common log, base 10
NOP		[Nop]	no-op: vacant program step
OP		[OP]	special operation
OP*		[OP] [Ind]	special operation indirect
P→R	(P−>R)	[P►R]	polar to rectangular conversion
PAR	(PART)	[Part]	partition (may be set indirectly)
PAU	(PAUSE)	[Pause]	pause
PI	(π)	[π]	3.14159265359
PRD	(PROD)	[Prd]	store the product of display and memory in memory
PD*	(PROD*)	[Prd] [Ind]	product indirect
PRT	(PRINT)	[Prt]	print
RAD		[Rad]	select radian units for angles
R/S		[R/S]	run/stop
RCL		[RCL]	recall from memory
RC*	(RCL*)	[RCL] [Ind]	recall indirect
RST		[RST]	reset program pointer
RTN	(RETN)	[INV] [SBR]	return to the step following the subroutine call

SBR		[SBR]	call subroutine (may be addressed indirectly)
SIN		[sin]	sine
ST		—	program step before 000, STart of program memory
ST.F	(STF)	[StF]	set flag (may be specified indirectly)
STO		[STO]	store in memory
ST*	(STO*)	[STO] [Ind]	store indirect
SUM		[SUM]	store the sum of display and memory in memory
SM*	(SUM*)	[SUM] [Ind]	sum indirect
TAN		[tan]	tangent
TRC	(TRACE)	[Trace]	trace each operation of a program or calculation
[$\sqrt{x}$]	($\sqrt{X}$)	[$\sqrt{x}$]	square root
1/X		[1/x]	reciprocal
|X|	(|X|)	[|x|]	absolute value
$\overline{x}$		[$\overline{x}$]	determine the mean value of entered data points
X/2	(X^2)	[x^2]	square
X⇄T	(X⇄T)	[x⇄t]	send the display number to the t register and bring the t register number to the display
X=T		[x=t]	test whether the display is equal to the t register (The destination may be addressed indirectly.)
X≥T		[x≥t]	test whether the display is greater than or equal to the t register (The destination may be addressed indirectly.)
Y/X	(Y^x)	[y^x]	raise the display number to a power

Notes to the TI-58/58C/59 User

If you already use a TI-58, TI-58C, or TI-59 calculator, you already know the programming language of the TI-66. However, a few differences are present in the TI-66.

Batteries: You can normally get a year's use from a set of batteries. There is no need for recharging and there is no AC adapter.

Display: The TI-66 display has angle mode indicators. For an error condition, Error with all decimals on is displayed. The decimals replace the error flash of the TI-58/58C/59. Fractions are displayed with a leading 0 so numbers with magnitude less than 1 are shown with a maximum of 9 digits.

Learn mode: You see the instruction you just keyed. It is inserted automatically after the displayed step. There is no need for an insert key. There is no "write-over" action. Delete has an automatic backstep.

Key codes: The 2-digit codes repesenting instructions have been replaced with mnemonics. This brings about an editing difference on the TI-66. When an instruction that uses an address is deleted but the address fragment remains, the TI-66 interprets these as No-ops, even though it appear as digits in the display. Internally, there are key codes but these do not correspond to the row-column arrangement of the keyboard.

Partitioning: The TI-66 has two methods. You can partition in groups of ten memories or you can partition for any number of memories between 0 and 64.

Program Entry: The TI-66 can only receive programs from the keyboard. There is no card reader and there is no module port.

Printer Connection: The cord connection solves the dilemma of having to remove the battery pack (and losing the memory) to attach the calculator to the printer. Simply plug the connector into the side of the calculator any time you need the printer.

Printing: The PC-200 has a 16-character line. OP codes one through five are slightly different on the TI-66. The alpha register quarters now cover four characters each. Also, the TI-66 alpha codes correspond to different characters than the TI-58/58C/59 alpha codes.

Items in TI-58/58C/59 Programs Which Require Alteration to Run on the TI-66

A program that calls a library program will not work on the TI-66. You need to write your own subroutine to replace the library program.

Whenever a program is to use any of the printer OP codes, it must be changed to conform to the OP codes of the TI-66.

The TI-66 replaces the error flash with all decimals on except the true decimal.

Comparisons are made on values as they are displayed and not all their internal digits.

From the keyboard, a miskeyed arithmetic operation can be corrected immediately by pressing the desired operation key. Consecutive operations result in the last one being used, not an error.

Rectangular to polar conversion yields angles from $-180°$ to $+180°$ instead of $-90°$ to $270°$.

Input range to trig functions is $\pm 4999999999°$ instead of $\pm 9.999999 \times 10^{99}°$.

DSZ IND works only on memories 0—9.

The number copy operation using [CE] after an open parenthesis no longer needs the [CE].

LBL cannot serve as a label.

IND handles numbers no larger than 9.999999×10^{12}.

[INV] [2nd] [Σ+] does not subtract 1 from the t register.

There are no HIR commands or other hidden features on the TI-66 that you may have accessed on the TI-58/58C/59 through illegal key sequences.

INDEX (Also see the Key Index inside the front cover.)

NOTES

ONE-YEAR LIMITED WARRANTY

THIS TEXAS INSTRUMENTS ELECTRONIC CALCULATOR WARRANTY EXTENDS TO THE ORIGINAL CONSUMER PURCHASER OF THE CALCULATOR.

WARRANTY DURATION This calculator is warranted for a period of one year from the date of the original purchase by the consumer.

WARRANTY COVERAGE This calculator is warranted against defective materials or workmanship. THIS WARRANTY IS VOID IF THE CALCULATOR HAS BEEN DAMAGED BY ACCIDENT, UNREASONABLE USE, NEGLECT, IMPROPER SERVICE OR OTHER CAUSES NOT ARISING OUT OF DEFECTS IN MATERIALS OR WORKMANSHIP.

WARRANTY DISCLAIMERS ANY IMPLIED WARRANTIES ARISING OUT OF THIS SALE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED IN DURATION TO THE ABOVE ONE-YEAR PERIOD. TEXAS INSTRUMENTS SHALL NOT BE LIABLE FOR LOSS OF USE OF THE CALCULATOR OR OTHER INCIDENTAL OR CONSEQUENTIAL COSTS, EXPENSES, OR DAMAGES INCURRED BY THE CONSUMER OR ANY OTHER USER.

Some states do not allow the exclusion or limitation of implied warranties or consequential damages, so the above limitations or exclusions may not apply to you in those states.

LEGAL REMEDIES This warranty gives you specific legal rights, and you may also have other rights that vary from state to state.

WARRANTY PERFORMANCE

During the above one-year warranty period, your calculator will be repaired or replaced with a new or reconditioned unit of the same or equivalent model (at TI's option) when the unit is returned by prepaid shipment to a Texas Instruments Service Facility listed below. In the event of replacement with a reconditioned model, the replacement product will continue the warranty of the original calculator or six months, whichever is longer. Other than the postage requirement, no charge will be made for the repair or replacement of in-warranty units.

Texas Instruments strongly recommends that you insure the unit for value, prior to shipment.

TEXAS INSTRUMENTS CONSUMER SERVICE FACILITIES

U.S. Residents	*Canadian Residents*
Texas Instruments Service Facility 2303 North University Lubbock, Texas 79415	Geophysical Services Incorporated 41 Shelley Road Richmond Hill, Ontario, Canada L4C5G4

Consumers in California and Oregon may contact the following Texas Instruments offices for additional assistance or information.

Texas Instruments Consumer Service 831 South Douglas Street El Segundo, California 90245 (213) 973-1803	Texas Instruments Consumer Service 10700 Southwest Beaverton Highway Park Plaza West Beaverton, Oregon 97005 (503) 643-6758

IMPORTANT NOTICE REGARDING PROGRAMS AND BOOK MATERIALS

TEXAS INSTRUMENTS MAKES NO WARRANTY, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR PARTICULAR PURPOSE, REGARDING THESE PROGRAMS OR BOOK MATERIALS OR ANY PROGRAMS DERIVED THEREFROM AND MAKES SUCH MATERIALS AVAILABLE SOLELY ON AN "AS IS" BASIS.

IN NO EVENT SHALL TEXAS INSTRUMENTS BE LIABLE TO ANYONE FOR SPECIAL, COLLATERAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH OR ARISING OUT OF THE PURCHASE OR USE OF THESE BOOK MATERIALS OR PROGRAMS AND THE SOLE AND EXCLUSIVE LIABILITY OF TEXAS INSTRUMENTS, REGARDLESS OF THE FORM OF ACTION, SHALL NOT EXCEED THE PURCHASE PRICE OF THE CALCULATOR. MOREOVER, TEXAS INSTRUMENTS SHALL NOT BE LIABLE FOR ANY CLAIM OF ANY KIND WHATSOEVER AGAINST THE USER OF THESE PROGRAMS OR BOOK MATERIALS BY ANY OTHER PARTY.