Miscellany 12

This chapter will acquaint you with additional features of TUTOR and PLATO. See Appendix A for sources of additional information.

Other Terminal Capabilities

We have emphasized the keyboard and plasma display panel as the main input and output devices used in communicating with the student. Other devices which may be used include a projector of color photographs, a touch panel, a random-access audio playback device, and other specialized input-output devices. All of these terminal-associated devices are easily managed by TUTOR.

The plasma display panel is flat and transparent, which makes it possible to project photographs on the back, superimposing color photographs with plasma-panel text and line drawings. There exists a microfiche projector for the PLATO terminal which will project any of 256 color photos, with fractional-second access time to any of these 256 pictures. (A "microfiche" is a sheet of film carrying many tiny pictures.) Microfiches can be made from a set of ordinary 35mm slides. Students or teachers can insert the appropriate microfiche in the terminal for the subject to be studied. The -slide- command selects any of the 256 photos: "slide 173" will project the 173rd photo. Additional options on the

-slide- command permit the independent control of a shutter in the projector.

The touch panel is a device which puts a grid of 16 vertical and 16 horizontal infrared light beams just in front of the plasma panel. When a student points at the panel, he breaks a horizontal and vertical beam. The information as to which beams were broken is sent to the computer as a "key" and the lesson can use this information to move a cursor, choose a topic pointed at, etc.

We discussed in Chapter 8 how to know where the student touched the screen. Another way is to use the information in the system variable "key", which contains the last "key" input from the student, whether it came from the keyboard, the touch panel, or some other external input device. Here is a unit which will analyze the inputs:

```
unit    getkey
next    getkey
enable
pause
goto    (key $ars$ 8),x,keyset,touch,extin,x
write   Impossible!
unit    keyset
write   You pressed a key
        on the keyboard.
*
unit    touch
calc    x⇐(key $ars$ 4) $mask$ o17
        y⇐(key $mask$ o17)
write   You touched location
        x=⊲s,x⊳,y=⊲s,y⊳.
*
unit    extin
write   The external input
        was ⊲s,key $mask$ o377⊳
```

The -enable- command permits touch inputs as well as inputs from any device connected to the external input connector at the back of the PLATO terminal. (The external input device might be a temperature sensor, an analog-to-digital converter, etc.) Without an -enable- command these inputs are ignored. A -disable- command will also cause inputs to be ignored. The system variable "key" contains a 1∅-bit integer (see the section on bit manipulations in Chapter 9): the most significant or left-most two bits identify the source of the key (∅ for keyset, 1 for touch panel, 2 for external input), and the least significant or right-most eight bits contain the actual data (which keyset button, which touch panel beams, what external data). In the case of the touch panel, the eight data bits contain four bits of x and four bits of y to specify a position. A succession of external inputs can also be retrieved with a single -collect-command.

If an -enable- command is placed just after an -arrow-, touch inputs can be accepted. There is a -touch- judging command whose tag specifies a screen location (and optionally a spatial tolerance). The -or- command is particularly useful here:

```
.
.
arrow    2513
enable
touch    1215
or
answer   book
write    Yes, "libro" means book.
```

The student will get the same message whether he or she types "book" or points at a picture of a book displayed at location 1215. (The -or-command can be used to make synonomous any judging commands. The system variable "anscnt" will be the same for all judging commands linked by -or-.)

There is a random-access audio device which stores twenty minutes of speech, music, or other sounds. Segments as short as one-third second can be accessed in a fraction of a second, no matter where the segment is located on the twenty-minute magnetic disk. As with microfiche, students can change the disks themselves. There is a -play- command to choose a section of the disk to play music or talk to the student.

Other devices can be connected to the external output connector at the back of the PLATO terminal and controlled with the -ext- command. The -ext- command can send up to sixty 16-bit quantities per second to a device. Among the interesting devices using this capability is a "music box" that plays four-part harmony.

Student Response Data

A crucial aspect of TUTOR on the PLATO system is that student response data can be collected easily to aid authors in improving lessons. Detailed information can be collected: unanticipated "wrong" responses (which may have been correct but inadequately judged), requests for help, words not found in a -vocabs-, etc. Summary information can also be collected: amount of time spent in an area of a lesson, number of errors made, number of help requests, etc. These detailed and summary data provide an objective basis for revising lessons.

A -dataon- command in a lesson turns on the automatic data collection machinery. Students registered in courses with associated response data files will have their responses logged in their data files. When registering students in a course, specific data collection options can be chosen. For example, one might collect only responses judged "no" (unanticipated incorrect responses). Anticipated correct responses (judged "ok") and anticipated incorrect responses (judged "wrong") would not be logged. This is often done because the anticipated responses are precisely those for which the lesson is already replying in a detailed, appropriate manner to the student. Here we see the difference between judge "no" (unanticipated) and judge "wrong" (anticipated). In this connection, -wrong-, -wrongv-, and -wrongu- make a "wrong" judgment, whereas the -no- command makes a "no" judgment.

The -area- command is used to divide a lesson into sections, each of which will produce an area summary in the data file. Each time the student encounters another -area- command, a summary of the previous area is placed in the data file. The area summary includes student name, area name, amount of time spent in the area, number of -arrow-s, number of ok/wrong/no responses, number of helps requested and found, etc. This summary data makes possible a statistical treatment of lesson data which can pinpoint weak areas.

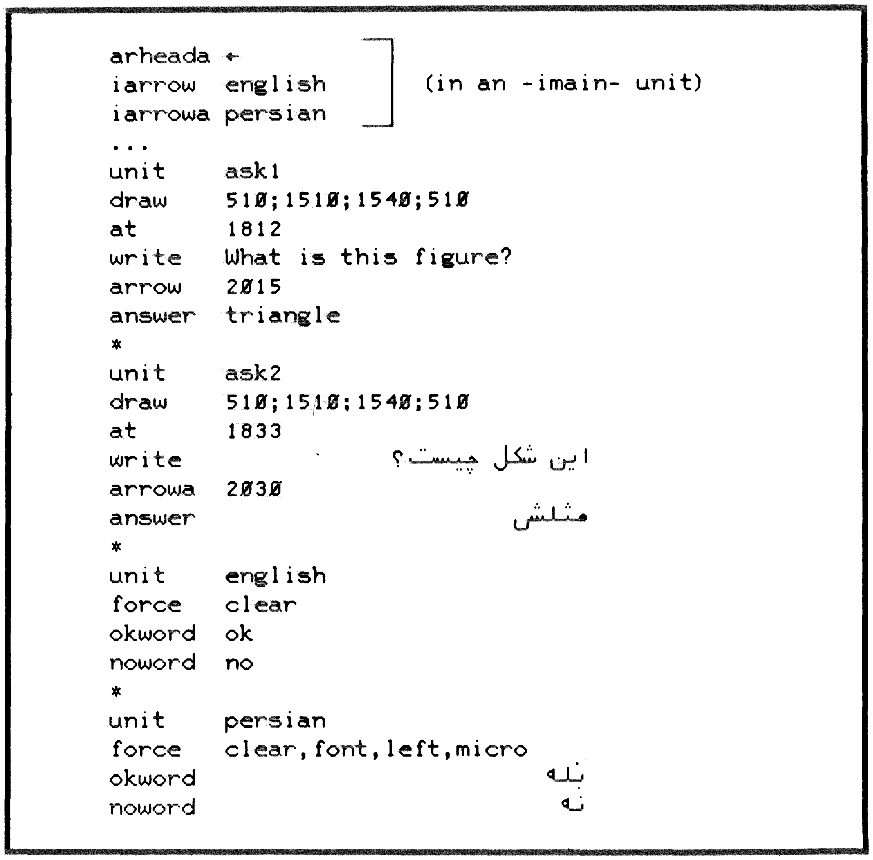
The -output- and -outputl- commands permit you to write your own information and messages into the datafile. This supplements the automatic data logging invoked with -dataon- and -area-.

While PLATO provides a standard mechanism for looking through data files (including sorting the data), you can also read back this information and process it yourself. For example, the -readd- command will read area summaries or -outputl- information from a datafile previously specified by a -readset- command.

Additional Tools for Teaching Foreign Languages

Usually in a lesson on a language such as Russian, which uses a special alphabet, the student will answer some questions in English and some questions in the foreign alphabet. The responses in the foreign alphabet require a "force font", or a "force font,left" for leftward-going languages such as Arabic, Hebrew, and Persian. Sometimes a "force micro" option will also be required in order to re-order the keyboard. Since there may be several things different about the two kinds of -arrow-s, it is convenient to have an alternate -arrow- command, which is named -arrowa-.

The -arrowa- command can cue the student differently, because you can alter the arrowhead displayed by -arrowa- by using the -arheada- command. The -arheada- command is similar to the -okword- and -noword- commands (the tag is what will be shown). Just as an -iarrow- unit is associated with the -arrow- command, so the -iarrowa- command can be used to specify a unit associated with the -arrowa- command. Here is a typical setup:



```
arheada ←          ┐
iarrow  english    │   (in an -imain- unit)
iarrowa persian    ┘
...
unit    ask1
draw    51Ø;151Ø;154Ø;51Ø
at      1812
write   What is this figure?
arrow   2Ø15
answer  triangle
*
unit    ask2
draw    51Ø;151Ø;154Ø;51Ø
at      1833
write                        این شکل چیست؟
arrowa  2Ø3Ø
answer                                مثلث
*
unit    english
force   clear
okword  ok
noword  no
*
unit    persian
force   clear,font,left,micro
okword                                 بله
noword                                  نه
```

Fig. 12-1.

Unit "ask2" has an -arrowa- command, which is associated with unit "persian", the unit named in the earlier -iarrowa- statement. Unit "persian" clears out any existing -force- options and sets up the appropriate typing conditions for the student. Unit "persian" also redefines the words to be shown for correct and incorrect responses. The -answer- command in unit "ask2" has the Persian for triangle. The student will see a "←" instead of a " ⩾" as a cue to give a response, thanks to the -arheada- command. On the other hand, the standard -arrow- command in unit "ask1" has associated with it the -iarrow- unit "english", which clears the -force- options and sets the "ok" and "no" words to English words.

While this machinery is particularly valuable in language lessons, it is also useful whenever your -arrow-s fall into two rather different categories. An example might be a physics lesson in which some -arrow-s require sentence responses and other -arrow-s require algebraic or numerical responses.

Some additional TUTOR commands which are particularly helpful in foreign language lessons include -change-, -getword-, -getloc-, -getmark-, and -compare-. As an example of the -change- command, the statement "change symbol comma to word" (which must be placed in the initial entry unit) will change the normal meaning of a comma as an ignorable punctuation mark, so that the comma will be treated as a separate word. This is useful when teaching punctuation, where you want to check specifically for commas. The -getword- command is similar to -storen- and is used to pull apart the student's response into separate words. The -getloc- command will tell you where a particular student word is on the screen, so that you could draw a box around that word. The -getmark- command gives you information on how TUTOR marked up the student's response, including whether a word was incorrect, misspelled, or out of word order. The -compare- command permits you to check a student's word against a stored list of words (in a common, for example), including spelling aspects.

Routers and -jumpout-

A lesson can be designated as a "router" which routes students through the many lessons making up a complete course. A router is associated with a course. Students registered in a course which uses a router will upon sign-in be sent first to the router, not to the lesson specified by the restart information. A typical router might ask the student, "Do you want to resume studying the lesson you last worked on?" If the student says yes, the router executes a "jumpout resume", which means "jumpout" of this lesson into the lesson mentioned in the tag, with "resume" having the special meaning "resume at the restart point". If the student does not want to resume, the router might offer the student an index of available lessons. Suppose the student chooses a lesson on the list whose name is "espnum". Then the router does a "jumpout espnum" to take the student to that lesson. (The -jumpout- command can be conditional.) Upon completion of lesson "espnum", (by "end lesson") the student is brought back into the router. If the lesson executed a -score- command, the router can use the corresponding value of system variable "lscore" to help decide how to route the student. The router might ask the student what he or she wants to do next, or the router might immediately take the student to an appropriate lesson.

Generally speaking, -jumpout- commands should be placed only in routers, not in instructional lessons. Following this practice insures that lessons can be plugged into routers on a modular basis. An exception exists in the case where one instructional package is spread over two or three physical lessons, in which case -jumpout- is used to connect the lessons.

A router can use up to fifty "router variables" (vr1 through vr5Ø) which are not affected by the instructional lessons. These can be used to keep track of which lessons have been completed, how many times they have been reviewed, how much time was spent in each lesson, etc.

Instructor Mode

Authors write and test lessons, and students study these lessons. Instructors choose lessons from the library of available lessons to make up a course for their students. Instructors also register students, monitor their progress, leave messages for the class or for individual students, etc. There is an "instructor mode" which makes it easy for instructors to do these things without knowing the TUTOR language. The instructor mode is based on a router coupled with a mechanism for setting up a roster of students. The options available through this router are sufficiently flexible to make it unnecessary in most cases to write specialized routers.

Special "terms"

Authors have a number of special "terms" to help them in curriculum development. If you press TERM and type "step", you can step through your lesson one command at a time. (A continued -calc- counts as one command.) This is extremely helpful in tracking down logical errors in a lesson. After each step, you can check the present value of student variables. There is also a -step- *command* which will throw the lesson into the step mode. The step features are operative only for authors testing their own lessons.

"TERM-cursor" provides you with a cursor which you can move around the screen using the "arrow" keys. Press "f" for fine grid or "g" for gross (coarse) grid. Also press "f" or "g" to update the display of the current cursor location. This facility is useful for deciding what changes to make in the positioning of displays on the screen.

"TERM-consult" notifies PLATO consultants of your request for help. When a consultant becomes available, he or she will talk to you by typing at the bottom of your screen. The consultant's screen has the same display you have on your screen. It is as though the consultant were looking over your shoulder as you demonstrate the problem. You can talk to the consultant by typing sentences at -arrow-s or by hitting TERM and typing. (If you press NEXT, and you have typed eight or fewer characters, your sentence will be taken as a -term- to look for in the lesson. Otherwise your line is erased so that you can type some more.) The consultants not only know TUTOR well but they have also had a great deal of experience in helping authors.

"TERM-talk" asks you for the name of the person you want to talk to, then pages that person if the person is presently working at a PLATO terminal. The person called accepts the call by hitting TERM and typing "talk". The two of you can then talk to each other at the bottom of the screen, but neither of you can see what is on the rest of the other person's screen. If you want the other person to see all of your screen, press shift-LAB, which puts you into a mode similar to TERM-consult.

"TERM-calc" provides a convenient one-line desk calculator at the bottom of the screen. Authors get normal, octal, and alphanumeric results. To avoid confusion, students who use TERM-calc are not shown the octal and alphanumeric displays.

Appendices

Appendix A. Where to get further information

Appendix B. List of TUTOR commands

Appendix C. List of built-in -calc- functions

Appendix A
Where to Get Further Information

The document "Summary of TUTOR Commands and System Variables" by Elaine Avner lists each TUTOR command, gives the basic form of the tag, and notes any restrictions such as maximum number of arguments or maximum length of names. Lesson "aids" available on PLATO provides detailed interactive descriptions of each command, as well as a wealth of other information useful to authors.

Lesson "notes" on PLATO provides a forum for discussing user problems. You can write notes to ask questions or to suggest new features that would be helpful in your work. You can read notes written by other users, including replies to your notes. Replies to programming questions generally appear within one day. (For faster service, use TERM-consult.) An extremely important section of "notes" is the list of announcements of new TUTOR features. Check this section regularly for announcements of new TUTOR capabilities. The announcements are followed within a few days by detailed descriptions in "aids".

Sometimes "notes" will announce a *change* in the TUTOR language involving an automatic conversion of existing lessons. For example, at one time there were several different commands (-line-, -liner-, -figure-, and -figuref-) which did what -draw- now does. When -draw- was implemented, all existing PLATO lessons were run through an automatic conversion routine to change the old commands into appropriate -draw- commands. It is probable that other such refinements will be made in the future. Therefore, be sure to read notes and aids regularly.

Appendix B
List of TUTOR Commands

Display

- at,atnm
- write
- writec
- erase
- eraseu
- size
- rotate
- mode
- charset
- lineset
- micro
- char
- plot
- show
- showa
- showe
- showo
- showt
- showz

- draw
- box
- vector
- circle
- circleb
- dot
- window

- rorigin
- rat,ratnm
- rdraw
- rbox
- rvector
- rcircle
- rdot

- gorigin
- axes
- bounds
- scalex
- scaley
- lscalex
- lscaley
- labelx
- labely
- markx
- marky
- gat,gatnm
- graph
- hbar
- vbar
- gdraw
- gbox
- gvector
- gcircle
- gdot
- polar
- delta
- funct

- slide
- play
- ext

Calculations

- calc
- calcc
- calcs
- define
- do
- exit
- doto
- goto
- branch
- transfr
- zero
- set

- randu
- setperm
- randp
- remove
- modperm

- pack,packc
- move
- search
- compute
- itoa
- clock
- name
- course
- date
- day
- find
- findall

- common
- comload
- storage
- stoload
- initial
- reserve
- release
- sort
- sorta

Sequencing

- unit
- entry
- nextnow
- next,next1
- back,back1
- help,help1
- data,data1
- lab,lab1
- term
- base
- end
- restart

- imain
- finish

- do
- join
- exit
- goto
- jump
- jumpout

- eraseu

- nextop,next1op
- backop,back1op
- helpop,help1op
- dataop,data1op
- labop,lab1op
- termop

Student Responses

- arrow,endarrow
- iarrow
- arrowa
- iarrowa
- long
- jkey
- copy,edit
- force

- answer,wrong
- answerc,wrongc
- concept,miscon
- vocabs,vocab
- list,endings
- ansv,wrongv
- ansu,wrongu
- exact,exactc
- touch,touchw
- ok,no,ignore
- ans
- match
- specs
- or

- storea
- storen
- store
- storeu

- judge
- join

- bump
- put,putd,putv
- loada

- okword,noword

- eraseu

- getword
- getloc
- getmark
- compare
- change

Other

- pause
- catchup
- time
- step
- keytype
- group
- collect

- inhibit
- enable
- disable

- dataon
- area
- output
- outputl
- readset
- readd

- dataset
- datain
- dataout

Additional TUTOR Commands Not Discussed in This Book

abort	abort normal updating of common or student record
add1	add one to a variable
allow	allow an instructional lesson to use router common
altfont	use alternate font for all writing
backgnd	run lesson at lower priority
chartst	check whether charset already loaded
close	like -loada- but takes one character per variable
dataoff	turn off student response data collection
delay	timed blank output for precise display timing
exactv	character string match to student response
foregnd	run lesson at normal (non-background) priority
iferror	specify unit to go to if -calc- error
lesson	sets "ldone" to inform router about lesson completion
open	like -storea- but stores one character per variable
press	presses a key for the student
readr	read a student record for data processing
record	record a message on audio device
route	specify router units for end of instructional lessons
routvar	set up router variables
stop	like -back- but for the STOP key
sub1	subtract one from a variable
tabset	set up tabs for TAB key
timel	set a time within a lesson
timer	router sets a time for a lesson to finish
use	use sections of another lesson to prepare this lesson

Appendix C
List of Built-in-Calc-Functions

sin(x)	sine
cos(x)	cosine
arctan(x)	arctangent

Angles are measured in radians. For example, sin(45) means sine of 45 radians, but sin (45°) means sine of 45 degrees (Ø.7Ø7). The degree sign (MICRO-o) converts to radians. Similarly, arctan(1) is .785 radians, which can be converted to degrees by dividing by 1°, the number of radians in one degree; arctan(1)/1° is 45. Using the degree sign after a number is equivalent to multiplying the number by (2π/36Ø). π (MICRO-p) is 3.14159. . . .

sqrt(x)	square root; can also be written $x^{1/2}$ or $x^{.5}$
log(x)	logarithm, base 1Ø
ln(x)	natural logarithm, base e
exp(x)	e^x
abs(x)	absolute value; abs(−7) is 7
round(x)	round to nearest integer; round(8.6) is 9
int(x)	integer part; int(8.6) is 8
frac(x)	fractional part; frac(8.6) is Ø.6
sign(x)	+1 if x>Ø, Ø if x=Ø, −1 if x<Ø
=,≠,<,>,≤,≥	produce logical values (true=−1,false=Ø)
not(x)	inverts logical values (true↔false)
x and y	true if both x and y are true
x or y	true if either x or y is true (or both)
x cls y	circular left shift x by y bit positions
x ars y	arithmetic right shift x by y bit positions
x $mask$ y	sets bits where both x and y have bits set
x $union$ y	sets bits where either x or y has bits set (or both)
x $diff$ y	sets bits where x and y differ (exclusive union)
bitcnt(x)	counts bits

The logical operators (=, ≠, <, >, ≤ and ≥) consider two quantities to be equal if they differ by less than one part in 1Ø^{11} (relative tolerance) or by an absolute difference of $1Ø^{-9}$. One consequence is that all numbers within $1Ø^{-9}$ of zero are considered equal. Similarly, "int" and "frac" round their arguments by $1Ø^{-9}$ so that int(3.999999999) is 4, not 3, and frac(3.999999999) is Ø, not 1. This is done because a value of 3.999999999 is usually due to roundoff errors made by the computer in attempting to calculate a result of 4.

System Variables

DISCUSSED IN THIS BOOK

anscnt
args
clock
formok
jcount
key
opcnt
spell
station
varcnt
vocab
where
wherex
wherey

NOT DISCUSSED IN THIS BOOK

baseu
capital
dataon
entire
error
errtype
extra
judged
ldone
lscore
lstatus
mainu
mode
nhelpop
ntries
order
phrase
size
user
wcount
zreturn

aarea
aarrows
ahelp
ahelpn
aok
aokist
asno
aterm
atermn
atime
auno

The third column consists of counters associated with the -area- command.

There are some additional system variables available for special purposes. See the on-line PLATO aids for information.

Index

- -abort- Appendix B
- absolute graphics commands 189, 190
- accent marks 10
- ACCESS key 175, 182
- active lesson 239
- -add1 - Appendix B
- aids Appendix A
- algebraic and numerical judging 126
  - algebraic 128
    - judging equations 131
    - warning about (1/2×) 132, 135
  - with scientific units 133
    - warning about (3+6cm) with -storeu- 135
- -allow- Appendix B
- alphanumeric information
  - -storea- 104
  - -showa- 53, 104
  - 10 characters per variable 105, 156, 162, 220
  - difference from numeric 105, 220
  - alphanumeric to numeric conversion 231, 235
- alternate font 175
  - unaffected by -size- and -rotate- 179
  - using -char- and -plot- 199
- -altfont- Appendix B
- and (and) logical operator 81
- And(array) 216
- Anderson, B. 4
- animations 28
  - use of iterative -do- 49
  - smooth animations 178
- -ans- 154
- anscnt system variable 113
  - zeroed when judging starts and by -specs- 113
  - zeroed by judge rejudge 120
  - not changed for synonomous -concept-s 116
  - cursor moving 122
  - with -or- 251
- -ansu- 135
  - warning about (3+6cm) with -storeu- 135
- -ansv- 126
  - -wrongv- 126
  - in arithmetic drill 127
  - with opcnt 127
  - specs noops,novars 128
  - concept/vocabs similar to ansv/define 128
  - algebraic judging 128

-ansv- *(Cont.)*
    warning about (1/2×) 132
  affected by -bump-, -put-, and judge rejudge 232
-ansva- Appendix B
-answer- 16
  markup of errors in student response 17
  with numbers 106
    Limitations 106
    notoler, nodiff 107
  with phrase (Santa*Maria) 17
  specs 107
  caps in tag with specs okcap 107
  no punctuation marks in tag 108
    punctuation ignored in student response 108
  with -list- 110
  -answer- useful in limited context 111
    see -concept- 111
  interaction with -concept- 114
  with negation 125
  with blank tag 126
  -exact- compared with -answer- 136
  conditional -answer- (-answerc-) 137
  using -put- to find pieces of words 159
-answerc- 137
Arabic 177, 253
-area-252
args system variable 55
arguments
  passing arguments to TUTOR commands 53
  passing arguments to subroutines 53
    args system variable 55
    warning to use different variables in different subroutines 56
    omitted arguments 55
    order of passing 54
    passing arguments in conditional -do- 79
    passing arguments in -goto- 90
  can be complicated expressions 55
arc of a circle 26
-arheada- 253
arithmetic drill 127
arithmetic right shift ars 224
  with negative numbers 228
arrays 214 (also see indexed variables 204)
  array operations 216
    matrix multiplication (dot product), vector product, sum, Prod, Min, Max, And, Or, Rev, Transp

arrays *(Cont.)*
  offset arrays 217
  vertically segmented arrays 231
arraysegv 231
-arrow- 15, 96 (also see -arrowa- 253)
  multiple -arrow-s in a unit 21, 98
  displays arrowhead on screen 16, 141
    inhibit arrow 122
  location in unit remembered 96, 141
  restarting at -arrow- for each response 97, 141
  satisfy all -arrow-s before leaving main unit 97, 142
  search for additional -arrow-s 97, 99, 142
    -goto- skipped 147
  delimits preceding -arrow- 97, 99, 142
  changes search state to regular state 99, 142
  sets default long 104
  summary of processing stages 141
    interactions with other commands 149
    sets default long, jkey, copy 150
  rules for attaching units containing -arrow- 100, 148
  merely collect response 164
  sets left margin 172
  with response erasing 192
  -enable- for touch input 250
-arrowa- 253
  different arrowhead from -arheada- 253
  associated -iarrowa- 253
assignment of values in a -calc- 46
  multiple assignments 47
  implicitly defined 203
  in -store-/-compute- 235
  specs okassign 235
assignment symbol 46, 235
asterisk for comments 20
attached unit 40, 64, 86
  by -do- 40
  by -goto- 86
attempts (counting student attempts) 119
audio device 251
automated display generation 35
automatic response-associated erasing 193
automatic scaling with graphing commands 182
    auxiliary unit (see attached unit)
Avner, E. Appendix A
-at- 14, 24 (also see -atnm- 172) (-gat- 183, -rat- 189)
  default -at- after response 97

-at- *(Cont.)*
  one or two arguments 24, 53, 55
  sets left margin 171
    -atnm- does not set a margin 172
  where system variable 173
  wherex and wherey system variables 174
-atnm- (-at- with no margin) 172 (-gatnm- 189, -ratnm- 189
-axes- 183 (also see -bounds- 184)

-back- 18 (also see -backop- 73)
-back1- 69 (also see -back1op- 73)
-backgnd- Appendix B
BACK and BACK1 return from help sequence 62
-backop- 73 (also see -back- 18)
  alternative to "inhibit erase" 73
-back1op- 73 (also see -back1- 69)
  alternative to "inhibit erase" 73
backspace 9, 174
-base- 63
  base pointer and base unit 63
    q or blank to clear 64
    automatically cleared when base unit reached 64
    set base pointer 63, 64, 198
base unit 19, 63
basic TUTOR 13
binary notation 209, 218
bit manipulation 217
  cls circular left shift 223, 224
  ars arithmetic right shift 224
    with negative numbers 228
  $mask$ 225
    constructing masks in octal 226
  $union$ 229
  $diff$ 229
  bitcnt function 229
  packing data 225
  octal numbers 226
  complementing bits 228
  byte manipulation 229, 230
bitcnt function 229
-bounds- 184
-box- 25 (-gbox- 185, -rbox- 189)
-branch- 212 (also see -goto- 85, -doto- 213)
  statement labels 212
  must not have duplicate labels 212
  cannot branch past -entry- 212
  speed advantage compared with -goto- 212

branching 59
  conditional 77
  within a unit, see -branch- 212
broken or dashed circle -circleb- 26
-bump- 120, 156
  combinations of -put- and -bump- 158
  with shift characters 158
  affects -store-/-ansv- 232
bumpshift specs option 109
byte manipulation 229, 230 (also see bit manipulation)

calc special term 256
-calc- 46, 201
  conditional -calc- (-calcc- and -calcs-) 84
    × is not the fall-through option 84
  summary 201
  statement label equivalent to -calc- 212
  with integer variables 222
  functions Appendix C
-calcc- 84 (see -calc-)
-calcs- 84 (see -calc-)
calculations 43
carriage returns and left margins 171
-catchup- 32
central memory 241
-change- 108, 136, 254
changes in TUTOR Appendix A
character grid
  coarse 14
  fine 23
character set 176
character strings 159
  see -bump- and -put- for student character strings
  see -pack-, -move-, and -search- for other strings
  single quote marks ('dog') 160, 223
  double quote marks ("dog") 165, 223
  6-bit character codes 220
  and -calc- 221
  and -compute- 231
characters
  character grid (coarse 14, fine 23)
  character size (8×16) 34
  10 per variable 105, 220
  special characters 175
-char- 199
-charset- 176, 181
-chartst- Appendix B
charts (see graphing commands)
Cheshire cat 40

Chinese characters with -rdraw- 188
-circle- 26 (-gcircle- 185, -rcircle- 189)
  ellipses 185, 189
-circleb- 26
circular left shift cls 222, 224
clear (force option) 253
-clock- command 163
clock system variable 163
-close- Appendix B
coarse grid 14, 23
command 13
  list of commands Appendix B
comments (*) 20, ($$) 26
-comload- 243, 247
-common- 237
  temporary common 237
    uses of temporary common 239
  -common- not executed 238
  permanent common 240
    splitting among many students 248
  and the swapping process 240
  reserving common 246
common variables 237 (also see -common-)
-compare- 254
compile 231
complementing bits 228
-compute- 231 (see -itoa- 235)
conditional commands 77 (also see -if- 91)
  condition can be complicated expression 79
  condition rounded to nearest integer 79, 80
  with logical expressions 80
    more precise due to rounding 80
consult special term 256
continued -write- statement 171
conversions
  between octal and decimal 226
  between alphanumeric and numeric 231, 235
-course- 163
  course registration 199
-concept- 111 (see -vocabs- 111)
  with numbers 113
  with numbered vocabulary words 117
  synonyms 111
  with phrases (Santa*Maria) 116, 118
  with endings 116, 118
  markup of student response 113
    missing words 113
    misspellings 113
  specs okextra 113
-concept- *(Cont.)*
  interaction with -answer- or -wrong- 114
  with judge wrong 115
  synonomous -concept-s 116
    anscnt unchanged 116
  with negation 125
  concept/vocabs similar to ansv/define 128
-copy- 150, 10
  copy key disabled by -arrow- 150
  copy compared with edit 150
cross product (vector product) 216
cursor moving routine 122
  with -match- 124
  with -keytype- 166
cursor special term 256
Curtin, C. 5
Cyrillic characters 176

dashed or broken circle (-circleb-) 26
data from student responses 251
-data- 69 (also see -dataop- 73)
-dataoff- Appendix B
-dataon- 252
-dataop- 73 (also see -data- 69)
data bases 237
data files 252
-datain- 248
-dataout- 248
dataset operations 248
-dataset- 248
-data1- 69 (also see -data1op- 73)
    -data1op- 73 (also see -data1- 69)
-date- 163
Davis, C. 2
-day- 163
debugging facilitated by -do- 42
decimal and octal conversions 226
-define- 47, 202, 235
  use -define-, avoid primitives 48
  -define- must precede related -calc- 47
  explicit multiplication required 48, 235
  overriding system variable definitions 56, 235
  student define set 103, 128, 231, 235
    with algebraic judging 128
    with scientific units 133
    with indexed variables 205
    in grafit 234
  defining functions 202
  warning about defining v, n, vc, or nc 205

-define- *(Cont.)*
  defining arrays 214
  defining indexed variables 205
  defining segmented variables 207
-delay- Appendix B
-delta- 185
desk calculator 101
dialog (with -concept- and -vocabs-) 111
dictionary using -term- 71
$diff$ 229
dimensionality of scientific units in -storeu- 133
-disable- 250
disk permanent storage 241, 242
-do- 40 (also see -doto- 213, -if- 91)
  iterative 49
    compared with conditional -goto- 85
    caution about slowness of segmented variables 211
  conditional 78 (also see -if- 91)
  conditional iterative -do- 90
    special meaning of q and x 91
  undo when -unit- command encountered 87
  do q like goto q 89
    like -join-, except regular only 98, 142
  skipped during judging and search 98
  do-ing -arrow-s 100
  text-insertion nature 101
    -goto- causes exception 87, 145
  judging command prevents un-do-ing 142, 145
  do level saved at -arrow- 149
  nested -do-s 206
  -exit- from -do-s 236
dollar signs for comments 26
-dot- 200
-doto- 213
dot product (matrix multiplication) 216
dots on screen 24
  -dot- 200
display screen 3
displays 23
  automated display generation 35
-draw- coarse grid 14, fine grid 25
  automated display generation 35
  example with complicated expressions 52
  from current position 174, 186
  skip option 185
  updating of where, wherex, wherey 185
-draw- coarse grid *(Cont.)*
  large number of points 186
  comparison with -gdraw- 189, 190
  comparison with -rdraw- 189, 190
  see -window- 190
  erasing associated with response 195
  making dots 200
drills
  arithmetic 127
  vocabulary 137, 138, 196

-edit- 150, 9
  edit compared with copy 150
ellipses (-gcircle- 185, -rcircle- 189)
-else- 91
-elseif- 92
embedded show commands in -write- statement 53
    s for -show-, a for -showa-, t for -showt-, e for -showe-, and z for -showz- 53
  in -writec- 84
  in -pack- and -packc- 162
-enable- 250
-end- 19, 63, 64, 198
  ignored in non-help sequence 64
  end lesson 139, 255
  no -end- with -helpop- 73
-endarrow- 21, 99, 100
  delimits preceding -arrow- 21, 100, 142
  changes search state to regular state 100, 142, 148
  pause between -arrow-s 100
  at end of unit 100
  required if -arrow- done or joined 100, 148
-endif- 92
-endings- 116, 117, 118
-entry- 89 (also see -unit-)
  use in vocabulary drill 197
equality rounding
  in logical expressions 81
equations in algebraic judging 131
-erase- 28, 33
  automatic full-screen erase for new main unit 22, 60
    inhibit erase 151, 197
    explicit -erase- 198
  -nextop- alternative to "inhibit erase" 73
erase mode 33
  used in erasing responses 196
-eraseu- 195

erasing student responses 192
-exact- 136
  handles punctuation marks 136
  blank -answer- not blank -exact- 126
-exactc- 136 (conditional -exact-)
exclusive union (see $diff$ 229)
-exit- 236
exponential show command, -showe- 53
exponents in floating-point numbers 219, 229
expressions (mathematical) 43
  usable everywhere 52
  logical expressions 80
    mixing logical and numerical expressions 81
  student expressions 101, 102
-ext- 251
external input 250
external output 251

false (in logical expressions) 80
-find- 235 (also see -search- 161)
-findall- 236
fine grid 23
-finish- 239
flags using segmented variables 210
floating-point numbers 219, 229
font 11, 175
  force font 177
  unaffected by -size- and -rotate- 179
  using -char- and -plot- 199
-force- 104
  force clear 254
  force long 104
  force font 177
  force left 177, 253
  force micro 253
-foregnd- Appendix B
foreign languages 137, 196, 252
formok system variable 102, 129, 134, 233
-funct- 185 (also see 233)
function keys 9
functions 48, 202, Appendix C
  parentheses around function arguments 48, 102
  dimensionless arguments for -storeu- 134
  defining your own functions 202
  int (integer part) 204, 222
  sin (sine) 48
  sqrt (square root) 52
  modulo 204
  bitcnt (bit count) 229

functions *(Cont.)*
  plotting functions 185, 233
  array functions 216

-gat- 183 (also see -at-)
-gbox- 185
-gcircle- 185
-gdraw- 183 (also see -draw- and -rdraw-)
  comparison with -draw- 189, 190
  comparison with -rdraw- 189, 190
-getloc- 254
-getmark- 254
-getword- 254 (also see -storen- 125)
Ghesquiere, J. 2
-gorigin- 183
  comparison with -rorigin- 189
-goto- 85 (also see -branch- 212, -doto- 213)
  mild form of -jump- 85
  cut off a unit 85
  does not change main unit 85
  relation to -do- 85, 86, 87
  exception to text-insertion nature of -do- 87, 146
  summary of basic properties 88
  goto q 88, 139
  with -entry- 89
  compared with iterative -do- 90
  passing arguments with -goto- 90
  a regular command 98, 146
  skipped during judging and search 98
  must not use in attached -arrow- unit 100, 148
grafit language 234
-graph- 183
graphics 23
  automated display generation 35
  comparison of absolute, relative, and graphing 189, 190
graphing commands 182 -gorigin-, -axes-, -bounds-, -scalex-, -scaley-, -labelx-, -labely-, -lscalex-, -lscaley-, -markx-, -marky-, -gat-, -gatnm-, -graph-, -hbar-, -vbar-, -gdraw-, -gbox-, -gvector-, -gdot-, -polar-, -delta-, -funct-
grid 23
-group- 167 (see -keytype- 166)
  with touch panel 168
-gvector- 184

halfcirc subroutine example 46
-hbar- 183 (also see -vbar- 183

Hebrew 177, 253
-help- 18, 62 (also see -helpop- 72)
  later -help- overrides earlier help 21
-helpop- 72
  return to waiting point, not start of unit 73
  no -end- command 73
-help1- 69
-help1op- 73
help sequence 62 (also see -helpop- 72)
  help sequence is a slow subroutine 66
    return is to beginning of base unit 66
  converting between help and non-help sequences 64
  use of -jkey- to give help 152
  importance of enabling HELP key 154
  with inhibit erase 198

-iarrow- 75, 155, 253
-iarrowa- 253
ieu (see initial entry unit)
-if- 91
-iferror- Appendix B
-ignore- judging command 122
-imain- 73
inactive lesson 239, 241
indenting with -if-/-else- 92
index for students to use
  with -term- 70
  with -imain- 73
  setting and clearing -imain- 74
  with -store-/-ok- 103
  with -match- 125
  with -ansv- 126
indexed variables 204
  with -storeu- dimensionality 133
  warning about defining v, n, vc, or nc 205
  indexed common variables 238
-inhibit-
  inhibit arrow 122
  inhibit erase 151, 197
    interaction with -restart- 199
    -nextop- alternative to "inhibit erase" 73
initial entry unit (ieu) 177
  with compute pointers 234
  relation to -restart- 178, 199
initializations
  general questions of initialization 66, 67
  unit pointers cleared when new main unit entered 70
  use of -imain- 74
initializations *(Cont.)*
  zeroing variables 207
  zeroing compute pointers 232
    in ieu 234
  -window- not initialized by main unit 191
  -size- and -rotate- not initialized by main unit 190
  with -restart- 199
    initializing variables 199
  zeroing temporary common 238
  zeroing -storage- 247
insertion of subroutine (by -do-) 40
instructor mode 255
int function for integer part 204, 222
integer variables 221
  common integer variables 238
interactions of -arrow- with other commands 149
Introduction to TUTOR, Ghesquiere, Davis, Thompson 2
iterative -do- 49, 67
-itoa- 235

jcount system variable 105
  affected by specs bumpshift 109
    and -bump- 156
    and -put- 157, 159
-jkey- 150, 151
  default set by -arrow- 150
  with response erasing 192
-join- 98 (also see -do-)
  universally executed (regular, judging, search) 98, 142
  like -do- except universal 98, 142, 144, 155
  join-ing -arrow-s 99
  text-insertion nature 101, 145
    -goto- causes exception 87, 145
  judging command prevents un-do-ing 142, 145
  repeated execution in regular, judging, search states 142
-judge- 115, 118
  -judge- is a regular command 115, 118
  judge wrong used to stay at -arrow- 115, 197
    does not stop processing 119
      judge noquit does stop processing 119
    in student data 252
  judge ok 118
    does not stop processing 119

-judge- *(Cont.)*
judge okquit does stop processing 119
judge continue 119, 153
in algebraic judging 131
judge rejudge 120, 156
affects -store-/-ansv- 232
judge ignore 121
stops processing 121
judge exit 123
judge no 123
does not stop processing 119
judge noquit does stop processing 119
in student data 252
judge quit, okquit, noquit 123
conditional form of -judge- 118
judging commands 95
(see -arrow-, -answer-, -wrong-, -answerc-, -wrongc-, -concept-, -miscon-, -match-, -ansv-, -wrongv-, -ansu-, -wrongu-, -store-, -storea-, -storen-, -exact-, -exactc-, -ignore-, -ans-, -bump-, -put-, -putd-, -specs-, -endarrow-)
summary 139
stop processing in regular state 97, 144
may terminate judging state 97
ok and no judgments 97
default no 97
require an -arrow- command 96
skipped in search state 97, 98
delimit regular commands 98, 142
accessed by -join- 98
switching from regular to judging state 119
judging copy of student response 120
affected by -bump- 156
judging keys 150 (see -jkey-)
judging student responses 95
-jump- 68
initializations 68
base pointer not affected 68
cancels previous -do-s 68
screen erased 68
used with -base- to initiate help sequence 68
compared with -goto- 85
-jumpout- 254

key system variable 152, 165
key names 152, 165
catching every key 164
key codes 165
timeup 166
with touch and external input 250
keyset or keyboard 8
-keytype- 166 (see -group- 167)
with touch panel 168
keyword judging 123
-kstop- Appendix B

-lab- 69 (also see -labop- 73)
-labl- 69 (also see -lablop- 73)
labeling graphs 183
labels on statements for -branch- 212, for -doto- 213
must not have duplicate labels 212
-labelx- 183 (see -markx- 184)
-labely- 183 (see -marky- 184)
-labop- 73 (also see -lab- 69)
-lablop- 73 (also see -labl- 69)
languages 137, 196, 252
large-size writing 26
left shift (see circular left shift 222, 224)
leftward writing 177, 252
lesson samples 4-6
lesson space 181, 240
lesson not swapped 244
levels of -do- (10 permitted) 41
line drawings (see -draw-)
line-drawn characters (see -size- and -rotate-) 179, 188
-lineset- 179, 181, 188
-list- 110
in -answer- and -wrong- 110
-loada- 159, 160
locking common 246
logical expressions 80
in conditional commands 80
mixed with numerical expressions 81
logical operators =,≠,<,>,≤,≥ 80
roundoff on equality 81
logical operators and, or, (not) 81, 82
-long- 103
force long 104, 150
follows -arrow-, precedes judging commands 104
modifies -arrow- 104

-long- *(Cont.)*
  must precede -specs- 107
  long 1 with judge ignore 122
  default set by -arrow- 150
  -edit- for long greater than 150 characters 150
-lscalex- 184 (see -scalex- 183)
-lscaley- 184 (see -scaley- 183)
lscore (associated with -score-) 255

main unit 59, 64, 85
  not affected by -goto- 85
margin set by -at- and -arrow- 171
marker
  -arrow- marker 96, 97
  -specs- marker 109, 114
markup of response 97
-markx- 184 (see -labelx- 183)
-marky- 184 (see -labely- 183)
masking in bit manipulations ($mask$) 225
-match- 123
  also see -storen- 125
  in grafit language 234
mathematical expressions 43
matrix multiplication 216
matrix operations 214 (also see arrays)
Max(array) 216
merge (see $union$ 229)
-micro- 181
  force micro 253
microfiche 249
micro-key options 10
micro table 181
Min(array) 216
-mode- (erase, write, rewrite) 33, 174, 179
  conditional form 85
-modperm- 138 (also see permutations)
modulo function 204
-move- 160
multiple -arrow-s 21, 99
multiplication
  explicit between defined names 48 (except for students 103)
  takes precedence over division 44
music 251

-name- 163
naming variables (-define-) 47
nc1-nc1500 common variables 243
negative words 110, 125
-next- 18, 59 (also see -nextop- 73)
  put near beginning of unit 61
  successive -next- commands override 61
  “next ” or “next q” to clear pointer 61
NEXT key 9, 60
  always a judging key 150
  ignoring extra NEXT keys 155
next physical unit 60
-next1- 69, 70 (also see -next1op- 73)
-nextnow- 18, 20
-nextop- 73 (also see -next- 59)
  alternative to “inhibit erase” 73
-next1op- 73 (also see -next1- 69)
  alternative to “inhibit erase” 73
-no- 103, 123
  in arithmetic drill 127
nodiff specs option 107
non-help sequence 64
  converting between help and non-help sequences 64
non-numerical parameters specified by student 104
nookno specs option 108
noops specs option 128
noorder specs option 18, 108, 116
noquit (judge option) 119, 123
not (logical function) 82
notes Appendix A
notoler specs option 107
novars specs option 128
-noword- 197, 253
nr1-nr50 router variables 255
numbering vocabulary words 117
numeric information different from alphanumeric 105
  range of numerical values 217
numerical parameters specified by student 101, 126
  checking for negative 119
numerical and algebraic judging 126
  algebraic 128
n1-n150 student variables 221

octal numbers for masks 226
octal show command, -showo- 53, 227
offset arrays 217
-ok- 101, 119, 123
okassign specs option 235

okcap specs option 107
okextra specs option 18, 108, 113
okquit (judge option) 119 123
okspell specs option 107, 116
-okword- 197, 253
opcnt system variable 128, 129
-open- Appendix B
operations (see precedence)
optional words
  in -answer-/-wrong- 16
  in -vocabs- 111
-or- judging command 251
or (or) logical operator 81
Or(array) 216
-output- 252
-outputl- 252

-pack- 162
-packc- 162
parentheses around function arguments 48, 102
partial circle 26
passing arguments 53 (see arguments)
-pause- 28, 164
  between -arrow-s, with -endarrow- 100
  catching every key 164
  no key display 167
  no help at blank -pause- 167
  pause keys=a,b,etc. 168
    help, term, etc. possible 168
    NEXT key special 168
  with touch panel 168
permanent common 240 (also see -common-)
permanent storage area 240
permutations 138
  -randp- 138
  -setperm- 138
  -remove- 139
  -modperm- 138
  vocabulary drill 137
Persian 177, 253
photographic projection 249
phrase (such as Santa*Maria) 17, 116, 118
physical next unit 60
place notation 224
plasma display panel 3
-play- 251
-plot- 199
plotting functions 233 (also see -funct- 185)
pointers (next, help, base, etc.) 60
  q or blank to clear pointer 61, 65
  successive commands override earlier settings 61, 65
  cleared when new main unit entered 70
  compute pointer 232
    zeroing in ieu 234
pointing at touch panel 168, 250
-polar- 184
positioning 23
powers in floating-point numbers 219, 229
precedence (of mathematical operations) 44, 132
preparing lesson for active use 239
-press- Appendix B
primitive variable names (v1-v150) 44, 48, 235
Prod(array) 216
punctuation in responses 108, 126, 136, 254
-put- 120, 157
  affects jcount 159
  terminates judging if string too long 157
  combinations of -put- and -bump- 158
  affects -store-/-ansv- 232
-putd- 158 (also see -put-)
-putv- 158 (also see -put-)

q (special unit name) 61, 65
  clears unit pointers 70, 79
  goto q 88, 139
  in conditional iterative -do- 91
quit (judge option) 123
quote marks for character strings
  single ('dog') 160, 223
  double ("dog") 165, 223

random numbers (see -randu- and permutations)
-randp- 138 (also see permutations)
-randu- 82
  arithmetic drill 137
  algebraic judging 128, 129
  compared with -randp- 138
range of numerical values 217
-rat- 189
-ratnm- 189
-rcircle- 189
-rdraw- 187
  affected by -size- and -rotate- 188
  compared with -gdraw- 189

readability with subroutines 40
-readd- 252
-readset- Appendix B
records in datasets 248
-record- Appendix B
registration records 199, 242
  -storage- not saved 247
regular commands 96
  skipped in judging state 96, 120, 141, 146
  skipped in search state 97, 98, 142
  -do- and -goto- are regular commands 98
  switching from regular to judging state 119
  judging command stops and prevents un-do-ing 142, 144
relative graphics commands 189, 190
-release- 246, 248
-remove- 139 (also see permutations)
-reserve- (common 246, dataset 248)
reserving common 246
reserving dataset records 248
responses (see judging)
response data 251
-restart- 199 (also see initial entry unit 177)
  -storage- not saved 247
restarting a lesson 178 (-restart- command 199)
resume (in -jumpout-) 254
return from help sequence 63, 66
Rev(array) 216
rewrite mode 34, 174, 179
right shift (see arithmetic right shift 224, 228)
-rotate- 26
  interaction with -arrow- 149
  affects -writec- 84
  does not affect alternate font 179
  affects -rdraw- even in size zero 188
  not initialized by main unit 190
-rorigin- 187
  compared with -gorigin- 189
rounding
  of condition in conditional commands 79, 80
  in equality operation 81
  in indexed variables 205
  in segmented variables 211
  with integer variables 222
-route- Appendix B
routers 254, 255
router variables (vr1-vr50) 255
-routvar- Appendix B
Russian alphabet 176
-rvector- 189

-scalex-/-scaley- 183 (also see -lscalex-/ -lscaley- 184)
  comparison with -size- 189
scaling in graphing commands 182
scientific units 133 (see -ansu-)
-score- 255
-search- (character string command) 161
search state (looking for additional -arrow-s) 97, 142
  skips regular and judging commands 97
segmented variables 207, 230
  table of ranges and space 209
  signed segments 208
  fractional numbers 210
  slowness 211
  equivalent bit manipulations 225
  byte manipulations 229
  vertical segments 230
segmentv 230
selective erase (text 28, graphics 33)
sequencing 59
  summary of sequencing commands 69
  author-controlled and student-controlled 70
  within a unit, see -branch- 212
-set- (fill array elements) 217
-setperm- 138 (also see permutations)
Sherwood, B. 5
Sherwood, J. 7
shift character 104, 156, 157, 158, 162
shift operators (cls 222, 224) (ars 224, 228)
skip in -draw- 185
-show- 51
  significant figures 52
  -showa- (alphanumeric) 53, 105
    default length 105
    uses 6-bit character codes 220
    ignores null characters 222
    with v or n variables 222
  -showe- (exponential) 53
  -showt- (tabular) 53
  -showo- (octal) 53, 228
  -showz- (show trailing zeroes) 53
  automatic erasing 194

sign-in/sign-out 199, 242
simulation of judging and search 98
sin (sine function) 48
-size- 26
  interaction with -arrow- 149
  affects -writec- 84
  does not affect alternate font 179
  affects -rdraw- 188
  comparison with -scalex- 189
  not initialized by main unit 190
skipping over main units 59
-slide- 249
Smith, S. 4, 111
smooth animations 178
-sort- 248
-sorta- 248
sorting lists 248
special characters 175
specifying parameters
  numerical
    -store- 101
    with -show- 106
  non-numerical
    -storea- 104
    with -showa- 106
-specs- 17, 18, 107
    notoler, nodiff 107
    bumpshift 109
    okcap 107
    okspell 107
      with -concept- 116
    okextra 18, 108, 113
    noorder 18, 108
      with -concept- 116
    nookno 108, 115
    noops, novars 128
    okassign 235
  -specs- is a judging command 107
  -specs- sets a marker 109, 141
    later -specs- overrides earlier marker 109
  clears anscnt 114
speech 251
spell system variable 109
spelling and -compare- 254
square root function, sqrt(expression) 52
statement has command and tag 13
statement label with -branch- 212, with -doto- 213
  must not have duplicate labels 212
status bank 242
-step- command 256
step special term 255
-stoload- 247
-storage- 246 (also see -common-)
  not saved on sign-out 247
  zeroed on sign-in 247
-store- 101
  a judging command 102
  judges no if cannot evaluate 102
  with -show- 106
  compared with -storen- 125
  with -ansv- 126
  concept/vocabs similar to ansv/define 128
  warning about (1/2×) 132
  affected by -bump-, -put-, and judge rejudge 232
  no primitive variable names 235
  no assignments without specs okassign 235
store values into variables 44
-storea- 104
  with -showa- 106
  with character string manipulations 159
  opposite of -loada- 159
  compare with -pack- 162
  merely collect response 164
  uses 6-bit character codes 220
  with v or n variables 222
-storen- 126
  also see -match- 123 and -store- 101 and -getword- 254
-storeu- 133
  terminates judging if error 134
  warning about (3+6cm) with -storeu- 135
strings 159 (see character strings)
student define set 103 (also see -define-)
student responses 95
  storing responses (see specifying parameters)
  judging responses (see judging commands)
  student response data 251
student specification of parameters (see specifying parameters)
student variables (v1-v150) 44
  in displays 45
  compared with common variables 238
  augment with -storage- 246
-subl- Appendix B
Sum(array) 216
superimposing writing 34, 174

superscripts and subscripts 10, 174
system variable 55
  anscnt 113
  args 55
  clock 163
  formok 102, 129, 134, 233
  jcount 105
    affected by specs bumpshift 109
      and -bump- 156
      and -put- 159
  key 152, 165
  opcnt 128, 129
  spell 109
  varcnt 129, 132
  vocab 115
  where 173
    updating in -draw- 185
  wherex 174
  wherey 174
subroutines 39
superscripts and subscripts 174
swapping process 240, 243
  swapping memory 241
  and common variables 243
synonyms
  in -answer- 16, 95 (also see -list- 110)
  in -concept- 113 (also see -vocabs- 111)
  in numbered vocabulary words 118

table of square roots 52
-tabset- Appendix B
tabular show command, -showt- 53
tag 13
talk special term 256
temporary common 238 (also see -common-)
Tenczar, P. 6
-term- 70 (also see -termop- 72)
  complementary to -help- 71
  dictionary use 71
  duplicate terms an error 71
  synonyms 72
  step, cursor, consult, talk, calc 255
terminal capabilities 3, 249
-termop- 72
text (see -write-, -size-, -rotate-)
text insertion of subroutine (by -do-) 40
  -arrow- in subroutine 100, 148
Thompson, C. 2
tick marks on graphs 184
-time- 31
time-slice 245, 246
timeup key 166
tolerance
  with -answer-/-wrong- 107
  with -ansv-/-wrongv- 126
  with -ansu-/-wrongu- 135
  on equality operations 81
-touch- 251 (also see 168)
touch panel 168, 250
transfr- 207
  not with segmented variables 208
  with -common- or -storage- 247
Transp(array) 216
tries (counting student attempts) 119
true (in logical expressions) 80

unconditional commands 79 (also see conditional commands)
$union$ 229 (also see $diff$ exclusive union 229)
-unit- 14
  terminates preceding unit 87
    see -entry- (which does not terminate) 89
  must not have duplicate -unit- names 212
unit pointers (see pointers) 60
units (scientific units) 133
universal execution of -join- 98, 142
-use- Appendix B

varcnt system variable 129
variables
  student variables 44
    with -restart- 199
    with -storage- 246
  indexed variables 204
    with -storeu- dimensionality 133
  common variables 237
  segmented variables 207
  range of numeric values 217
  router variables 255
-vbar- 183 (also see -hbar- 183)
-vector- 25 (-gvector- 184, -rvector- 189)
vertical segments 230
vocab system variable 115
-vocab- 116
-vocabs- 111, 252 (see -concept- 111)
  numbering vocabulary words 117
vocabulary drill 137
vc1-vc1500 common variables 243
vr1-vr50 router variables 255

v1-v150 student variables 44
where system variable 173
  updating in -draw- 185
wherex system variable 174
wherey system variable 174
-window- 190
-write- coarse grid 14, fine grid 24
  with embedded show commands 53
    s for -show-, a for -showa-,
    t for -showt-, e for -showe-, and z for
      -showz- 53
  conditional -write- (-writec-) 82
  with left margins 171
  continued -write- statement 171
  successive -write- statements 172
  also see -size- and -rotate-
  size 1 versus size 0 188
  automatic erasing 192
  alternate font
    with charset 176
    using -char- and -plot- 199
write mode 34
-writec- 82 (also see -write-)
  x is not the fall-through option 83
  special character when using commas 83
  with embedded show commands 84
  affected by -size- and -rotate- 84
  automatic erasing 192
-wrong- 16 (also see -answer-)
-wrongu- 135 (also see -ansu-)
-wrongv- 126 (also see -ansv-)
  with scientific units 135

x (special unit name) 62, 79

-zero- 207
  not with segmented variables 208

$$ (permits comments to follow tag) 26
◁▷ embedding -show- in -write- 53