

STRUCTURED PROGRAMMING WITH COMAL - HOW TO USE IT WITH C64 COMAL 0.14

Structured Programming with COMAL was written using CBM PET COMAL 0.11 and Piccolo COMAL as examples. Both COMALs have since been upgraded, but since COMAL is standardized, few things have changed that affect the book. This page of notes is your key to adapting the book.

* Page 15: There is no ESCAPE key - use RUN/STOP key instead in most cases. Also no REPEAT key.

* SELECT OUTPUT "LP" now requires a colon after the LP=> SELECT OUTPUT "LP:" (see page 21)

* SELECT OUTPUT "DS" now requires a colon after the DS=> SELECT OUTPUT "DS:" (see page 21)

* BYE is not implemented - use BASIC instead (see page 22)

* RANDOMIZE is not needed in Commodore COMAL - omit it (see page 22)

* CHR(...) can be typed in but will be listed as CHR\$(...) - (see page 23)

* Variable names canNOT use SHIFTED letters. Only unshifted letters, numbers, and ']' (see page 26). Variable names can be up to 78 characters long. (Keywords cannot be used as a variable name: beware of: RIGHT, LEFT, BACK, HOME, RUN, KEY)

* Integer variables end with the # sign, not the % sign (see page 27)

* Substring specification has changed (see page 28 & 86) - now specify start and end character, not start character and number of characters.

SHOWN: rec\$(19:6) CHANGE TO: rec\$(19:24)

SHOWN: letter\$(pos:1) CHANGE TO: letter\$(pos:pos) OR: letter\$(pos)

* VAL (see page 29) is not implemented in COMAL 0.14. A proc to do this is included on the COMAL Disk (integers only) and in the COMAL TODAY newsletter (floating point).

* STR (see page 30) is not implemented in COMAL 0.14 but can be achieved with a user defined proc.

* ZONE (see page 35) statement changed from ZONE:=<num> into ZONE <num>.

SHOWN: ZONE:=8 CHANGE TO: ZONE 8

* Some programs assume an 80 column screen / printout. They must be adapted to a 40 column screen (see page 37 & 73).

* The book was written before the CURSOR command was implemented in CBM COMAL version 2.00 and RC COMAL 2.0. Both use CURSOR(ROW,COL). This book uses CURSOR(x,y) - (see page 36). C64 COMAL 0.14 doesn't have a CURSOR command, but a user defined CURSOR PROC is included on the C64 COMAL System Disk (using ROW,COL as it should be). To adapt to the way this book uses CURSOR merely use the following CURSOR2 PROC when requested by the book (it calls the REAL CURSOR proc which is listed below as well:

PROC CURSOR2(x,y) PROC CURSOR(ROW,COL)

CURSOR(x,y) POKE 212,0 //quote mode off::PET use POKE 205,0

ENDPROC CURSOR2 PRINT CHR\$(19) // HOME CURSOR

FOR Y:=1 TO ROW-1 DO PRINT CHR\$(17), //CURSOR DOWN

FOR X:=1 TO COL-1 DO PRINT CHR\$(29), //CURSOR RIGHT

ENDPROC CURSOR

SHOWN: EXEC cursor(37,11) CHANGE TO: CURSOR2(37,11) OR: CURSOR(11,37)

* BUG IN page 77 PROGRAM: Add following line just after the last ELSE on the page:

FOUND:=FALSE

* Page 97: SHOWN: SELECT OUTPUT PRINTER CHANGE TO: SELECT OUTPUT "LP:"

* BUG IN page 119 PROGRAM: add to second program line: ;VOW:=0

* RND should be replaced by RND(0) -- (see page 129)

*** Function specification was changed. In the early COMAL versions, a procedure was used as a function, merely assigning a value to the function name. Now, a function has its own header and end: FUNC and ENDFUNC, and the value is RETURNED. (see page 136):

SHOWN: PROC ran CHANGE TO: FUNC RAN

y:=25.173*x+3.921 Y:=25.173*X+3.921

x:=y-INT(y) X:=Y-INT(Y)

ran:=x RETURN X

ENDPROC ENDFUNC RAN

* A label must now be followed by a colon (:) -- (see page 139):

SHOWN: again CHANGE TO: AGAIN:

* Page 157 PROGRAM BUGS: SENTINEL\$ was misspelled once as SENTINAL\$, and the 3rd last line:

SHOWN: FOR L:= ... CHANGE TO: FOR K:= ...

* Page 160 PROGRAM BUG: SHOWN: en:=pair-1 CHANGE TO: EN:=PAIR

* Page 173 PROGRAM BUG: SHOWN: READ(row,col) CHANGE TO: READ ROW,COL

* Page 174 PROGRAM BUGS: SHOWN: change(row,col):=1 CHANGE TO: CHANGE(ROW,COL):=0

* Page 179 PROGRAM BUG: SHOWN: DIM queue(8) CHANGE TO: DIM QUEUE(0:6)

* Page 185 PROGRAM BUG: COMP was misspelled once as CAMP

* Page 203 PROGRAM BUGS: BIRTH\$ was misspelled once without the \$ as BIRTH, and

SHOWN: year:= ORD(y\$(... CHANGE TO: YEAR:=(ORDY\$(...

* When writing a string to a file, CBM COMAL precedes it with a two byte length counter. Thus the total length written to disk will be two bytes more than the length of the string (see page 204):

SHOWN: OPEN 2 "textfile" RANDOM 6 CHANGE TO: OPEN FILE 2,"textfile",RANDOM 8

Page 205 SHOWN: ... RANDOM 11 CHANGE TO: ... RANDOM 13

Page 206, 208, 210 SHOWN: ... RANDOM 17 CHANGE TO: ... RANDOM 19

* When writing a number to a file, CBM COMAL writes it in binary - always 5 bytes long, regardless of the number (see page 205)

A disk with all (that could fit on one disk) programs, examples, and samples correctly adapted to C64 COMAL is available for \$19.95 from COMAL Users Group, U.S.A., Limited.