NAME

MASE intrinsic functions

SYNOPSIS

MASE provides the user with many intrinsic functions. These functions may be selected via a menu by calling the function *COMMAND-MODE* that is bound to the colon key (‘‘:’’) by default. All these functions can be bound to a key code via the *BIND* function.

Some functions may read the current cursor position and use this in establishing defaults. Others may modify the current cursor position. *MULT* is a numeric multiplying prefix that may modify some commands. To specify a *MULT* factor, just type in the digits before running the function. If *MULT* is not specified, it defaults to ‘‘1’’. Functions employing these values are noted.

NOTE: Due to the way *MULT* is implemented, it is not possible to *BIND* nor *MAP* the digits (‘‘0’’ through ‘‘9’’).

This list is in alphabetical order; functional groups of commands are noted in the other help files.

Most commands may be terminated before they finish by hitting C-C.

DESCRIPTION

@ABORT-COMMAND

Do nothing - use this to exit *COMMAND-MODE* with no effect.

ALIAS

Create an alias. You will be prompted for the name of the alias to create, and what it should alias to.

ALIAS-LOAD

Load aliases from an alias file.

ALIAS-SAVE

Save current aliases to a file.

ALIGN-ON-PATTERN

Given a pattern, insert gaps (either at the beginning of the sequence, or after the most recent period (anchor point)) to align all occurences of that pattern. No allowance is made to do this for just some subset of sequences (sigh!). If a given locus doesn’t have the specified pattern pattern, user will be notified, and the sequence will be untouched. Before the actual insertions are done, the user will be informed of the position to which the pattern will be aligned. Control-C out at this point if this is not what you want done.
See also: Anchor functions, *SQUEEZE-OUT-GAPS* (might be useful to clean up some messes!).

ALIGNMENT-MODE

Set up keyboard bindings and mappings for editing the *ALIGNMENT* of sequences: All letters are unbound. Minus sign inserts a gap. Keys ‘‘x’’ and ‘‘d’’ delete under the cursor. *PROTECT* is turned *ON*.

Alias for *MODE-ALIGNMENT*.

ALPHABETIZE-LISTS

Put function and variable lists into alphabetical order. This makes *INCREMENTAL COMPLETE* work better, as alphabetically close names appear adjacent to each other. Hierarchy of command nomenclature has been organized to simplify this.

NOTE: this is now the default situation.

ANCHOR-CHECK
: See if each sequence contains the same number of anchors +

ANCHOR-CREATE
: Create an anchor at the specified column (insert a ‘‘.’’ in each sequence at specified position) +

ANCHOR-DELETE
: Delete anchor at specified offset.

ANCHOR-STRIP
: Strip all anchors from all sequences (perhaps to execute some other sort of analysis ...)

ANCHOR-SYNCHRONIZE
: Insert gaps where needed to align all of the anchors. +

APPEND-TO-YANK
: Append the next *MULT* sequence positions to the Yank Buffer. +
The sequence is not modified. +

APROPOS
: Display short information about all commands, variables ... that contain a key string. I.E. running *APROPOS* for *CONSENSUS* would display information about all commands and variables that talk about *CONSENSUS.*

AUTO-INSERT
: The key typed will have the high bit stripped, and then be inserted |
MULT times at the current cursor position. This is useful only when bound to a key. If one binds **‘‘A’’** to *AUTO-INSERT,* then types **‘‘A’’,** an **‘‘A’’** will be inserted at the current cursor position.

AUTO-OVERTYPE
: The key typed will be replace the letter under the cursor. This is only useful when bound to a key. The old letter is deleted, so *PROTECT* will be referenced. +

BELL
: Ring bell and do nothing. This is used as the default binding for otherwise unbound keys.

BIND
: Bind a key to a specific function. First, one will be prompted to strike the key to be bound. Next, one will be prompted for the name of the command to bind it to via the *MENU* interface.

Callable from *TAKE* files as ‘‘BIND key-code function-name’’.

BIND-ALL-ALPHA
: Bind all alpha keys starting at some modification of "A" to some function. (*AUTO-INSERT, AUTO-OVERTYPE, INS-COMPENSATED-BACKWARD,* and *INS-COMPENSATED-FORWARD* would be the most logical functions.) Thus, saying "Bind-All-Alpha a Auto-Insert" would bind "a" though "z" to "Auto-Overtype", and saying "Bind-All-Alpha M-A" woul bind "M-A" though "M-Z" to "Auto-Overtype". +

NOTE: specifying anything other than a modification of the letter "A" is an error. +

It is valid to specify prefix characters... +

CHARACTER-SET
: Change the display character aliasing set. This allows one to view the sequences with an aliasing character set - i.e. display serines and threonines both as **‘‘t’’.** This *DOES NOT* affect the actual sequence, nor the *SEARCH* or *PATTERN-HIGHLIGHT* routines. It DOES affect *CONSENSUS* generation, and *HIGHLIGHT-DIFFERENCES.*

Callable from *TAKE* files as ‘‘CHARACTER-SET filename’’.

If *DISPLAY-ALTERNATE-CHARACTER-MAP* is disabled, the actual letters will be displayed. See *FILE FORMATS* for alias file format. +

Affects: *CONSENSUS*, basic display, *HIGHLIGHT-DIFFERENCES*, and *OUTPUT-ALIGNED* +

COLUMN-COMPOSITION

Report the frequency of all characters found in a given column

COLUMN-CORRELATION

Look at coordinated changes down columns.

References *INTERNAL VARIABLES ‘‘COLUMN-CORRELATION-FIXED-P’’*, *‘‘COLUMN-CORRELATION-THRESHOLD’’*, and *‘‘COLUMN-CORRELATION-MODEL-SEQ’’*

COMMAND-MODE

Select a function to run. This works via the menu interface. When you have typed enough of the command name to make it unique, it will be filled in automatically. If the variable *INCREMENTAL-COMPLETE* is set, the command will be filled in as far as possible. +

COMMENTS-EDIT

Edit the comment field of a locus. The editor to use will be read from your environmental variable **‘‘EDITOR’’.** This may be set via ‘‘EDITOR=your-editor-choice; export EDITOR’’ for the Bourne shell (sh), or via ‘‘setenv EDITOR=your-editor-choice’’ for the C-shell (csh), and may be set up in your ‘‘~/.profile’’ or ‘‘~/.login’’ (respectively) initialization files. If **‘‘EDITOR’’** is not set, *vi* will be used. You will be prompted for the number of the locus on which to operate (default to the current sequence). The file name will consist of the locus name with a number appended. Spaces in this file name will be converted to carets (‘‘^’’), since CCA-EMACS barfs for file names with spaces. When you are finished, instruct the editor to save your changes and exit. +
When the comments are read back in, a ‘‘;’’ will be placed at the beginning of each line if it isn’t already there.

COMMENTS-VIEW

Display comment lines from a set of sequences that contain a specified regular expression. The comments for the requested sequence will be passed to ‘‘grep -i string’’. The string you respond with may contain flags to ‘‘grep’’ (like ‘‘-v’’). Also note that if the pattern you want to search for has spaces, it must be surrounded by quotes.

CONSENSUS -

Generate a consensus sequence from a set of sequences. This consensus sequence will be displayed at the bottom of the window. The aliasing character set (see *CHARACTER-SET)* is referenced and used to generate the consensus. The base occurring most frequently down a column will be placed into the consensus sequence. In there is a tie, a question mark (‘‘?’’) will be placed in the consensus sequence. The variable *CONSENSUS-USE-GAP* affects the way the consensus routine looks at gaps (‘‘-’’). If this variable is *ON*, gaps will be treated as any other sequence element. If it is *OFF*, gaps will be ignored unless there are no other bases down that column. The variable *CONSENSUS-THRESHOLD* is also referenced. The most frequently occurring base must occur in at least this fraction of active sequence before the base will be placed in the consensus sequence. This variable defaults to ‘‘0.0’’.

Callable from *TAKE* files as ‘‘CONSENSUS range’’. If ‘‘range’’ is not specified, ‘‘1-$’’ will be used.

CONSENSUS-ALTERNATE

Generate and display a consensus sequence, as in *CONSENSUS* above, by a slightly different method. You will be asked to define a ‘‘model sequence’’, and an extra sequence set. The +

''Consensus'' sequence will be built as follows: for each position, the consensus will be the character from the master sequence if that character appears in at least *CONSENSUS-THRESHOLD* fraction of the ''other'' sequences specified. +

CONSENSUS-OFF
: Disable display of the consensus line, giving one additional line of sequence information.

COPY-TO-YANK
: First, clear out the Yank buffer. +
Then, copy the next *MULT* sequence positions into the Yank Buffer. The sequence is not modified. +

CREATE-LOCUS
: Create a new locus. The new locus can be ''All New'', copied from a ''Sequence in Memory'' (with the option to translate it (in either one or three letter code), or ''Imported'' from another sequence file. -
You will be asked to assign a new name for the locus, what file you would like to place it in, and where you would like to place it on the screen.

If you ask for a translation, you will be prompted for the frame (1, 2, or 3). +

If the variable *TRANSLATE-IGNORE-GAPS* is *ON*, gaps and anchor points will be stripped before doing the translation. The product peptide will be resynchronized with the nucleotide sequence. +

CURSOR-OTHER-WINDOW
: Position cursor to the equivalent position in the other window.

DEL-BACK
: Delete *MULT* positions *BEHIND* cursor. Most useful when typing in new sequence information.

References *INTERNAL VARIABLE ''PROTECT''. Mouse Functions.*

DEL-COLUMNS
: Delete multiple columns from multiple sequences (delete a column of positions). You will be prompted for:

Delete columns in which sequences
: Which sequences to act on (default to the current sequence),

Delete at which column
: At which position to do the deletions (default to the current cursor position

Delete how many columns
: How many positions to delete (default to *MULT)*

NOTE that *DEL-COLUMNS* observes the status of the variable *PROTECT*. If one attempts to delete non-gaps when *PROTECT* is set, you will be asked how to proceed. You may:

PRESERVE
: this sequence and continue to act on subsequent sequences,

OVERRIDE
: protection of this sequence - delete positions in this sequence regardless of the protection

ABORT the delete operation.

See Also: *Mouse Functions.*

DEL-COMPENSATED-BACKWARD
: Delete the character under the cursor, then compensate backward. +

DEL-COMPENSATED-FORWARD

Delete the character under the cursor, then compensate forward. +

DEL-GAP

Delete *MULT* positions *UNDER* cursor.

References *INTERNAL VARIABLE ‘‘PROTECT’’.*

See Also: *MOUSE FUNCTIONS*

DNA-MODE

Set up the keyboard bindings and mappings for editing nucleic acid sequences: The letters ACTGYRJKLMBDHVN, and X insert themselves. All other letters beep. Minus sign inserts a gap. *PROTECT* is turned *OFF*

Alias for *MODE-DNA.*

DYNAMIC-HIGHLIGHT

This function should be considered to be in "alpha-test". It will search for matches to a weight matrix. This is a SCORING method rather than a plus/minus method. (The regular expression handlers are plus/minus). This method can allow different weights for each letter to be associated with each position of the oligo. +

GRIPE

Send a complaint, question, comment, or observation to the *MASE* administrator. This is done via the system *mail (1)* command. Terminate your message with C-D.

HELP

View the MASE online documentation. This is displayed via *BROWSER.*

HIGHLIGHT-DIFFERENCES

Highlight positions where the display letter that differ from a specific sequence. The test is made between the ‘‘display’’ characters from *CHARACTER-SET,* and is not affected by *DISPLAY-ALTERNATE-CHARACTER-MAP.* |
The ‘‘model’’ sequence is specified in this command. If this variable is set to ‘‘0’’, letters differing from the *CONSENSUS* sequence will be highlighted. These highlights can be printed via *OUTPUT-ALIGNED.*

Alternate method of getting at *INTERNAL VARIABLE HIGHLIGHT-DIFFERENCES.*

HIGHLIGHT-LOAD

Load the highlight flags from a file. Current highlights will be discarded. NOTE - the highlights file MUST be syncronus with the sequences file. +

HIGHLIGHT-POSITION

Highlight *MULT* positions following cursor. Highlight type ‘‘1’’ will be used.

HIGHLIGHT-SAVE

Store the highlight flags to a file, for later use by *HIGHLIGHT-LOAD.* +

INS-COLUMNS

Insert multiple columns into multiple sequences (insert a column of gaps). You will be prompted for:

Insert columns into which sequences

Which sequences to act on (default to the current sequence),

Insert at which column

At which position to do the insertions (default to the current cursor position, and

Insert how many columns
How many gaps to insert (default to *MULT)*

See Also: *Mouse Functions.*

INS-COMPENSATED-BACKWARD
Insert the TYPED key (one that has been bound to the fuction) under the cursor, then compensate backward. This is similar to *AUTO-INSERT* +

INS-COMPENSATED-FORWARD
Insert the TYPED key (one that has been bound to the fuction) under the cursor, then compensate forward. This is similar to *AUTO-INSERT* +

INS-GAP
Insert *MULT* gaps at the current cursor position.

See Also: *Mouse Functions.*

INS-NEXT-CHARACTER
Read a key, then insert MULT characters of that key. +

INTERPRET-LINE
Type in a line in *TAKE* format that should be interpreted. This may be an easier way to get to *BIND* and *MAP* because it will allow you to use the *STRING-CONVERSION* conventions.

JUMP-TO-END-OF-SEQUENCE
Move cursor to *MULT* positions before the end of the current sequence.

JUMP-TO-LOCUS-NAME
Jump to a specific locus name (you will be prompted for the name of the locus you want to jump to). The locus will be selected by the menu interface.

JUMP-TO-LOCUS-NUMBER
Go to sequence numbered *MULT.*

JUMP-TO-POSITION
Move cursor to position *MULT* in the current sequence.

JUMP-TO-POSITION-ABSOLUTE
Move cursor to position *MULT* IGNORING GAPS in the current sequence. +

LITERAL-NEXT
The next character will not be remapped. See *MAP* in *INTRINSIC FUNCTIONS* for information on mapping.

LOAD
Read in another sequence file.

Callable from *TAKE* files as ‘‘LOAD filename’’.

References *INTERNAL VARIABLE ‘‘SILENT’’;* If *SILENT* is not set, the name and length of each locus will be displayed at the bottom of the screen.

LOAD-FUNCTION
Load an object file. This lets you link functions into *MASE* with out having to relink the entire source. The objects to be loaded are usually the output of a compiler, like the ".o" file from "cc -c file.c". If you are interested in this, please contact Don Faulkner. +

MAP
Set key remapping. You will be prompted for which key stroke to map, then the strokes that key should send. Terminate the list of strokes to send with ‘‘^@’’ (may be sent by ‘‘CTRL-SPACE’’ on some keyboards).

Callable from *TAKE* files as ‘‘MAP key-code string’’.

MARK-JUMP

Go to mark number *MULT.* Change both the active sequence and active column. The marks jumped to are those that have been set by *MARK-SET.* If the mark number |
MULT was never set, the cursor will not be moved. |

MARK-SET

Set mark number *MULT* to be the current sequence and column. You may return to this position later via *MARK-JUMP.*

META-NEXT

The next character will be ‘‘meta-sized’’ - it will have the high bit set. Think of this as an extra ‘‘shift’’ function.
NOTE you should investigate the *PREFIX-CHARACTER* function before making much use of +
META-NEXT. +

MODE-ALIGNMENT

Set up keyboard bindings and mappings for editing the *ALIGNMENT* of sequences: All letters are unbound. Minus sign inserts a gap. Keys ‘‘x’’ and ‘‘d’’ delete under the cursor. *PROTECT* is turned *ON.*

Alias for *ALIGNMENT-MODE.*

MODE-DNA

Set up the keyboard bindings and mappings for editing nucleic acid sequences: The letters ACTGUYRJKLMBDHVN, and X insert themselves. All other letters beep. Minus sign inserts a gap. *PROTECT* is turned *OFF*

Alias for *DNA-MODE.*

MODE-PROTEIN

Set up the keyboard bindings and mappings for editing amino acid sequences: The letters JOU beep; all other letters insert themselves. Minus sign inserts a gap. *PROTECT* is turned *OFF*

Alias for *PROTEIN-MODE.*

MOUSE-FUNCTIONS

Full discussion of the mouse is contained in another document.

MOVE-LOCUS-FILE

Move a sequence into another file (no provision for copy). You will be prompted for a) which sequences to move (default to the current sequence) and b) the name of the file to move them to. NO OUTPUT will occur until *SAVE* is called.

See Also: *CREATE-LOCUS* +

MOVE-LOCUS-POSITION

Move a locus to another position within the current file.

MOVE-TO-INSIDE-LEFT

If *MULT* is not specified, or set to ‘‘1’’, move the position under the cursor to the right margin of the left window. If *MULT* is specified and not set to ‘‘1’’, put column number *MULT* to the right margin of the left window.

MOVE-TO-INSIDE-RIGHT

If *MULT* is not specified, or set to ‘‘1’’, move the position under the cursor to the left margin of the right window. If *MULT* is specified and not set to ‘‘1’’, put column number *MULT* to the left margin of the right window.

Problem: it is not possible to put position ''1'' on the inside right margin via this mechanism.

MV-DOWN

Move cursor to same position *MULT* sequences down.

MV-LEFT

Move cursor to *MULT* base earlier in the current sequence. If the variable *STOP-AT-BORDERS* is set (not zero), then *MV-LEFT* will stop when it reaches a window border.

See *STOP-AT-BORDERS* in the *VARIABLES* discussion.

MV-RIGHT

Move cursor to *MULT* base later in the current sequence. If the variable *STOP-AT-BORDERS* is set (not zero), then *MV-LEFT* will stop when it reaches a window border.

See *STOP-AT-BORDERS* in the *VARIABLES* discussion.

MV-UP

Move cursor to same position *MULT* sequences up.

OUTPUT-ALIGNED

Generate an output of the aligned sequences. Provision is made for mapping strings depending on either the composition (i.e. print A, L, V, and I in ''green'', and Y, F, and T in ''yellow''), on the highlighted patterns, or on highlighted differences compared to a reference sequence (see *VARIABLE HIGHLIGHT-DIFFERENCES).* If you prefix the file name with an equals sign (''=''), the file will be searched for in the *MASE* directory.

OUTPUT-ALIGNED will display the number of the page being written. This is done to give the user something to watch lest he thinks *MASE* is idle. See ''MAPPING FOR OUTPUT'' in *FILE FORMATS* for information on the format of mapping files. This explains how to have different patterns show up differently in the output. The *TUTORIAL* document contains a section on *OUTPUT-ALIGNED.*

References the *INTERNAL VARIABLES ''OUTPUT-CHARACTERS-BETWEEN-VERTICAL-BREAKS'', ''OUTPUT-CHARACTERS-PER-LINE'', ''OUTPUT-LINES-BETWEEN-HORIZONTAL-BREAKS'', ''OUTPUT-LABEL-EACH-PAGE'', ''OUTPUT-POSITION'', ''OUTPUT-LINES-PER-PAGE'', ''DISPLAY-ALTERNATE-CHARACTER-MAP'', ''OUTPUT-USE-FORM-FEED'', ''OUTPUT-SUPPRESS-LOCUS-NUMBERS'',* and *''HIGHLIGHT-DIFFERENCES''*. to modify the output style. See the discussions in *INTERNAL-VARIABLES* for details.

WARNING: *OUTPUT* and the CRT display are independent. Every attempt has been made to allow *OUTPUT* to mimic as closely as possible the behavior of the screen. There are bound to be failures in this attempt.

OVERTYPE-NEXT-CHARACTER

Read a key, then overtype MULT characters of that key. +

PATTERN-HIGHLIGHT

Highlight all occurrences of a Regular Expression in all sequences. This is done via the **GNU-EMACS** regular expression handler. Use ''man regex-gnu'' for detailed information (also available from *BROWSER* in the *MASE* node.)

The *MULT* is used to specify which ''color'' should be used to highlight this pattern for *COLOR CURSES*, use under System V UNIX curses library, |
and for *ALIGNED OUTPUT.*

You will be asked if you want to clear the existing highlights. The default answer is

determined by *PATTERN-CLEAR-DEFAULT.* If the variable *PATTERN-ASK-CLEAR* is set to ‘‘*NO*’’, you WILL NOT be asked if the existing highlights should be cleared; if *PATTERN-CLEAR-DEFAULT* is ‘‘*ON*’’, then the existing highlights will be cleared; otherwise, they will be left. Sequences which had matches to the pattern specified will be indicated with an asterisk at the left margin of the display. See discussion in variables section about *HIGHLIGHT-MERGE-METHOD.*
PATTERN-HIGHLIGHT will display the sequence numbers as the pattern is highlighted. This is done to give the user something to watch lest he thinks *MASE* is idle.

References *INTERNAL VARIABLES PATTERN-IGNORE-CASE, PATTERN-IGNORE-GAPS, PATTERN-ASK-CLEAR,* and *PATTERN-CLEAR-DEFAULT.*

Callable from *TAKE* files as ‘‘PATTERN-HIGHLIGHT [pattern-number] pattern’’. Optional ‘‘pattern-number’’ replaces *MULT* from the interactive case.

The history lists for *PATTERN-HIGHLIGHT* and *SEARCH* are shared.

PATTERN-HIGHLIGHT-CLEAR

Clear existing highlights.

Callable from *TAKE* files as ‘‘PATTERN-HIGHLIGHT-CLEAR’’.

PREFIX-CHARACTER

The character specified will have a uniq entry in a "Prefix" table. Thus, if you bind "ˆX" to *PREFIX-CHARACTER,* you could then bind "ˆX<DEL>" to some other function than <DEL>. This provides a more flexible binding interface than "META-NEXT". You can nest prefix characters as deeply as you want: You could bind "ˆA" to *PREFIX-CHARACTER,* then bind "ˆAˆR" to *PREFIX-CHARACTER,* then bind "ˆAˆRa" to *OVERTYPE* ...

PROTEIN-MODE

Set up the keyboard bindings and mappings for editing amino acid sequences: The letters JOU beep; all other letters insert themselves. Minus sign inserts a gap. *PROTECT* is turned *OFF*

Alias for *MODE-PROTEIN.*

QUIT

Exit *MASE.* If you have made changes that have not been saved, you will be given the option to *SAVE.*

Beware: *QUIT* unmaps ‘‘N’’, ‘‘n’’, ‘‘Y’’, ‘‘y’’, <CR>, and <NL>. This is to guarantee that one will be able to respond correctly to the *QUIT* queries.

QUIT will verify that you really want to exit *MASE.* If you say ‘‘NO’’, you will resume editing. Remember that your mappings for ‘‘NnYy<CR><NL>’’ will be lost.

REFRESH

Refresh (replot) the screen; useful if the display gets barfed on.

RENAME-LOCI

Rename loci with the same name, appending a semi-colon (‘‘;’’) and a number where needed to make all names unique.

Callable from *TAKE* files.

RENAME-LOCUS

Assign a new name to a specified locus.

REPEAT

Repeat the specified macro-character *MULT* times. The character should be something that maps into a series of non-interactive functions. An example might be: *BIND* the key ‘‘period’’ (‘‘.’’) to ‘‘3ij3h3ik’’ (insert three gaps, go down, back three, insert three more gaps, then go back up). Then do: ‘‘3’’, ‘‘:REPEAT<CR>’’, then ‘‘.’’. This would execute the macro string 3 times. *REPEAT* will probably be most useful if you *BIND* it to a key, like ‘‘r’’; then, you would be able to do something like ‘‘3r.’’ - lots of key-strokes saved!

SAVE

Save all modified files (those with modified or rearranged sequences). The sequence will be written with *SEQUENCE-LINE-LENGTH* bases per line. The user name and time will be added as a note to the top of modified files and to the end of the comments of all modified sequences.

SAVE-CONSENSUS

Store the consensus sequence to a file in standard sequence format.

Callable from *TAKE* files as ‘‘SAVE-CONSENSUS filename’’.

See Also: *CREATE-LOCUS.*

SEARCH

Search for a **GNU-EMACS** Regular Expression, beginning at cursor position to the end of the current sequence. See ‘‘man regex-gnu’’ for detailed information on the syntax of the regular expression handler.

References *INTERNAL VARIABLES SEARCH-IGNORE-CASE* and *SEARCH-IGNORE-GAPS.*

The history lists for *PATTERN-HIGHLIGHT* and *SEARCH* are shared.

SEARCH-AGAIN

Search again for the pattern specified in the last *SEARCH* AHEAD of the cursor.

SEARCH-BACKWARDS

Like *SEARCH,* but searches from the current cursor position backwards.

SEARCH-BACKWARDS-AGAIN

Look for last pattern specified occuring behind the cursor.

SEQUENCE-SLAVE-SYNCHRONIZE

Synchronize gap placement of one sequence to that of another. This can be useful if one sequence is a slave of another, (as three-letter amino acids, and nucleic acid code), and they have somehow gotten out of synch.

SET-VARIABLE

Modify the environmental variables. The variable to set is specified via the menu interface. To see what a given variable in the menu is for, hit a question mark (‘‘?’’) for a short, online help message.

SHIFT-LEFT

Shift current window left by *MULT* position.

SHIFT-RIGHT

Shift current window right by *MULT* position.

SHIFT-LEFT-BOTH

Shift both windows left by *MULT* positions.

SHIFT-RIGHT-BOTH

Shift both windows right by *MULT* positions.

SHIFT-LEFT-BOTH-WIDE
Shift both windows left by *MULT* screen widths.

SHIFT-RIGHT-BOTH-WIDE
Shift both windows right by *MULT* screen width.

SHIFT-LEFT-WIDE
Shift current window left by *MULT* screen width.

SHIFT-RIGHT-WIDE
Shift current window right by *MULT* screen width.

SHOW-BINDINGS
Display current key-to-function bindings.

SHOW-FUNCTIONS
List the functions that are available.

SHOW-MAPPING
Display current key-stroke mappings.

SHOW-VARIABLES
Display current contents of variables.

SHOW-YANK-BUFFER
Display the current contents of the Yank Buffer: +
Report the number of sequence positions currently in the Yank Buffer, then as much of the +
sequence as will fit on one line. If there is more than one line of sequence, an ellipsis ("...") +
will appear at the end of the string. +

SIMILARITY
Generate a set of similarity scores between sets of sequences. You will be prompted for:

Query sequences
The set of sequences to form the query (default to the current sequence),

Run against which sequences
The set of sequences to form the library (default to the current sequence),

Start of region to compare

End of region to compare
The regions of the sequences that are to be used in the calculation

Output to file
Where should the output be stored.

Specify **‘‘SCREEN’’** to display and discard output.

Specify **‘‘HPLOT’’** to display the score histogram graphically

Specify a file name to put the output in a file

If the variable *SIMILARITY-IGNORE-SELVES* is set, comparisons of a sequence against itself will be ignored.

SIMILARITY-MATRIX-LOAD
Load in an alternate similarity matrix. The format is described in *FILE FORMATS.*

Callable from *TAKE* files as ‘‘SIMILARITY-MATRIX-LOAD filename’’.

SQUEEZE-OUT-GAPS
Squeeze out gaps that occur in each of a set of sequences. You will be prompted for the set of sequences to squeeze. *SQUEEZE-OUT-GAPS* will find which positions are gaps for ALL

the specified sequences, and delete them.

SQUEEZE-OUT-GAPS-RESTRICTED
Like *SQUEEZE-OUT-GAPS*, except user will be prompted for the sub-range of the sequences to squeeze (like between positions 100 and 150). +

STORE-ALL
Save mapping, binding, and variable status to a file, as other STORE functions.

STORE-BINDING
Store the current key bindings to a file (specify the file name) formatted so that it may be *TAKE* en.

STORE-HISTORY
Not activated. This will allow one to save the "history" lists.

STORE-MAPPING
Store the current key mappings to a file (specify the file name) formatted so that it may be *TAKE* en.

STORE-VARIABLES
Store the current variable values to a file (specify the file name) formatted so that it may be *TAKE* en.

TAKE
Read *MASE* commands from a text file. You will be prompted for the name of the file to read. If you prefix the file name with an equals sign (‘‘=’’), the file will be searched for in the *MASE* directory. Only a subset of the *MASE* functions are available to this interface. They are:

ALIAS +
ALIAS-LOAD +
ALIAS-SAVE +
ALIGN-ON-PATTERN +
ALIGNMENT-MODE +
ALPHABETIZE-LISTS
ANCHOR-CHECK +
ANCHOR-SYNCHRONIZE +
BIND key-code function-name
CHARACTER-SET character-set-file
CONSENSUS [range]
CONSENSUS-OFF
DNA-MODE +
HIGHLIGHT-DIFFERENCES +
HIGHLIGHT-LOAD +
HIGHLIGHT-SAVE +
LOAD sequence-file
MAP key-code string
MODE-ALIGNMENT |
MODE-DNA +
MODE-PROTEIN +
PATTERN-HIGHLIGHT [pattern-number] pattern-to-highlight
PATTERN-HIGHLIGHT-CLEAR
PROTEIN-MODE +
RENAME-LOCI
SAVE-CONSENSUS filename
SET-VARIABLE variable-name variable-value
SIMILARITY-MATRIX-LOAD +

TAKE mase-commands-file
UNBIND
UNBIND-ALPHA
UNMAP
UNMAP-ALPHA
WRITE-TO-FILE filename

UNBIND

Bind all key strokes to *BELL.* This will probably inactivate the cursors.

Callable from *TAKE* files.

UNBIND-ALPHA

Same as UNBIND, but only works on alphabetic characters a-z and A-Z.

UNDO-EDIT

Undo the most recent CHANGE. Changes which can be undone are: inserts, deletes, create new sequences, zap sequences. (NOTE - functions like "INSERT-COLUMNS" and such do their work by executing multiple simple inserts). This is a new function, and should be considered in "alpha-test".

UNMAP

Remove all key mappings. This will probably inactivate the cursors.

Callable from *TAKE* files.

UNMAP-ALPHA

Same as UNMAP, but only works on alphabetic characters a-z and A-Z.

UN-HIGHLIGHT-POSITION

Remove highlight from *MULT* positions following cursor.

VERIFY

If the key typed matches the letter under the cursor, the cursor advances and nothing else happens. If the key doesn't match the letter under the cursor, *MASE* will beep. If the variable *VERIFY-PAUSE* is set, the user would then have to hit <CR> to continue. This is useful only when bound to a key.

WINDOW-SYNCHRONIZE

Align the ''other'' window to the ''active'' window (the one where the cursor is).

WRITE-TO-FILE

Write a specified set of loci to a specified file. The loci will NOT have their modified flags reset. This may be useful if you are trying to retrieve a set of loci from some larger data base file.

WRITE-REGION-TO-FILE

Like *WRITE-TO-FILE,* but only writes out a specified region.

YANK-LAST-DELETE

Insert contents of Yank Buffer at current cursor position. The Yank Buffer is filled in by three different mechanisms:

COPY-TO-YANK
Copys sequence into the Yank Buffer.

APPEND-TO-YANK
Appends sequence to the Yank Buffer.

All functions that delete positions. *MASE* will attempt to concatenate adjacent deletes into one yank. +

There is only ONE Yank Buffer; it is cleared by *COPY-TO-YANK* and all non-adjacent deletions. +
+

ZAP-SEQUENCES
Delete an entire locus.

FILES

SEE ALSO

FILE FORMATS

BUGS

Are you kidding? OF COURSE!