Search, replace, and insertion strings may be entered by keyboard or clipboard as normal characters, or as ASCII character codes.

ASCII is an acronym for American Standard Code for Information Interchange which defines the first 128 characters on computers that support the standard.   Since American (and most European) computers use 256 characters, that leaves 128 characters that can vary.   These are often called the extended ASCII, or extended, character set.

ASCII character codes can be used to describe any of these 256 characters to A/S.   Simply enter a caret ("^") character followed by the ASCII number of the desired character.   A carriage return character, for example, could be specified by entering "^13" in a text edit field.   Any character from 1 to 255 may be specified using up to three digits.

When ASCII codes can be used, the Special pop-up menu is available.   To help determine ASCII codes for specific characters, or to find specific characters, select the Show Chart… item from the pop-up menu.   A window containing a character chart will open.   Type a character or click on a character in the chart; the character name and corresponding ASCII code is shown in the window title.   Double-click a character in this chart to insert that character into the current edit text field.

In addition to the ASCII numeric codes, A/S follows Microsoft Word's convention of "^t" to describe a tab character and "^p" for a carriage return.

Leading zeros are acceptable in the ASCII number code since A/S scans up to three characters to the right of a caret.   Sometimes the use of leading zeros is required.   For instance, when the desired search string is a tab character (ASCII value of 9) followed by a number 1, a string of "^91" will not work since A/S interprets this as the character "[" which has a ASCII value of 91.   Instead, use the string "^0091" to cause the 009 to be interpreted as a tab and the 1 as the   "1" character or take advantage of the Word-style tab code and use the string "^t1" which is shorter and may be clearer.

Character codes can be automatically entered for tabs or CRs or escape characters by pressing the tab or return or escape keys while the command key is down; codes of ^t, ^p, ^27 are generated.   If these characters are not allowed in the current text field, they are not generated.

When specific characters are unknown, searches can employ several types of wildcard codes in Replacement Strings, Line Replacement Strings, and Leading Char strings.   Since wildcards are used for matching characters, strings that are used for insertion, such as BOF and EOF insert strings, do not allow wildcards.

A wildcard is specified by entering a caret followed by a type descriptor.   Use the code "^?" for match any character, "^#" for any digit character (0-9), "^*" for any letter character (a-z, A-Z, international characters), "^+" for any upper case character, "^-" for any lower case character, "^!" for any punctuation character, and "^@" for any character sequence.

The characters that are considered matches for the any digit, any letter, any upper case, and any lower case wildcards can be adjusted using the Edit Add/Strip application.

The any character wildcard ("^?") can be used in the Change To string in Replacement and Line Replacement strings.   Doing so means "put the character that's at the matched position in the text into the replacement string."

The any character sequence wildcard ("^@") can be used in both the Find What and Change To strings in Replacement and Line Replacement strings to replace characters on either end of the matched sequence.   However, it can only be used in the Change To string when it is also used in the Find What string.   When it is used in both strings, there are three implied character substrings; the matched characters prior to the characters represented by the ^@ code, the characters represented by the ^@ code, and the matched characters following the characters represented by the ^@ code.

Here are some examples of search strings that use wildcards:

Find What:	Change To:	Old Text:	New Text:
^44^032	^?•	Jon, 12, 42	Jon,•12,•42
^*,^?^#	^?;	Jon, 12, 42	Jon; 12, 42
,^?	^?•••	Jon, 12, 42	Jon,•••12,•••42
^?^*^?	**	Jon, 12, 42	**, 12, 42
^#^#	^?	Jon, 12, 42	Jon, 1, 4
^#		Jon, 12, 42	Jon, ,
^@	[^@]	Jon $boldW.	Jon [$bold]W.
$^@	/^@	Jon $boldW.	Jon /boldW.
^@	[]	Jon $boldW.	Jon []W.

In addition to the previous examples, Line Replacement Strings allow empty strings to be used as the Find What string to insert strings and relax the requirement of specifying characters before and after any character sequence ("^@") wildcards.   Both behaviors are only available when the Find Where pop-up menu is set to the At Beginning or At End and the Change How pop-up menu is set to Replace String.

Here are some examples that assume the Change How pop-up menu is set to Replace String (^32 is the ASCII code for the space character):

What:	Where:	To:	Old Text:	New Text:
	Beginning	•	This is text.	•This is text.
	End	•	This is text.	This is text.•
^@^32	Beginning	•^32	This is text.	• is text.
^@^-	Beginning	•	This is text.	•is is text.
^32^@	End	^?•	This is text.	This is •
^32^@	End	^32•^@	This is text.	This is •text.

Another code that can be used in replacement and line replacement search strings is the "^:" sub-search code.   This is not a wildcard since it does not represent any specific underlying characters.   It is used to specify a secondary search string and a search domain within which a sub-search is to be performed rather than the normal search domain of the entire file.   The any characters wildcard will typically be used to help define the search domain.

Characters preceding the sub-search code specify the search domain while characters following the code specify the sub-search string.   The string "<PRE>^@</PRE>^:^p" would match all CRs found in a section of the file that starts with "<PRE>" and ends with "</PRE>."   The any characters wildcard is typically used to define the search domain.

Here are some simple examples of how the sub-search code could be used:

Find What:	Change To:	Old Text:	New Text:
J^@2^:^*		Jon, 12, 42	, 12, 42
J^@2^:^?	•	Jon, 12, 42	•••••••, 42
^@:$	\	Jon $boldW.	Jon \boldW.

Because the caret is used to indicate ASCII codes and wildcards, using it in a search string can be tricky.   There are two simple rules to keep in mind.   First, a caret-caret ("^^") sequence is always interpreted as a single caret character.   Second, a single caret is always interpreted as a caret character unless it precedes a digit (0-9) or a "t", "T", "p", "P", "?", "#", "*", "+", "-", "!", "@", or ":" character.