

Robert Glaser N3IC
3922 Algiers Road
Randallstown, Md 21133
July, 1977

FMT - A Text Formatting Program

FMT permits text files to be written without care to the line lengths. Probably the greatest power that the formatter has is its ability to fill text. The process of filling amounts to creating an output with the maximum number of words per line without exceeding the specified maximum line length. Crudely, when FMT is filling text, it is listening to the typewriter bell for you, and returning the carriage when necessary without bothering you with the details. FMT will not permit words to be broken when satisfying this criterion, so the maximum number of words will be placed on each line, which will depend upon the length of the first word in the next line. FMT continues to fill until one of several things happens. Filling will be halted on the current line and continued on the next when a tab is found at the start of a line. This means a new paragraph, and of course that line must start on a new line. An empty line will also have the same effect. FMT can also be commanded to break filling at any point as described under the commands. FMT will also keep track of the number of lines printed, and go to a new page when needed. Headers and page numbers can also be generated.

When filling, there are two possible modes of operation. These are adjusting the right margin, and not. If the mode is set so the right margin is not adjusted then the output appears just as a secretary would type it (except there are no mistakes). When the right margin adjust option is enabled, however, FMT will always pad the typewritten line with spaces to force the last word in the line to end exactly at the right margin. FMT adds spaces from the right to the left; that is, single spaces already in existence will be made into double spaces from the right to the left, and if the right margin is still not reached, the double spaces will then be converted to triple spaces from the right to the left. This process continues until the margin limit is satisfied.

To use FMT, all that is required is to provide it with

the text as well as the necessary commands. FMT is quite versatile, and learning all of the commands is useful. A command line is specified by a line in the input text beginning with a period (.). Any such line in the input text will not be translated to the output text. Even if the line is not a valid FMT command, it will not be passed along. Valid commands are always two letters after the period. They may be upper or lower case letters. There may be additional arguments after the command, and these must follow immediately with no intervening spaces. Anything on the command line after the command will be ignored, so nonprinting comments may be placed on the command lines if desired.

Here are the FMT commands:

.fi - This is the fill command. When this command is encountered, filling will begin until it is disabled by the next command. The default is .fi - that is, if not specified, it is as if a .fi command had been there. It is generally only needed after filling has been halted, to tell FMT to once again fill.

.nf - The no fill command turns off filling. It can only be changed by the .fi command. It is generally used to directly translate some lines in the input text directly through to the output.

.br - The break command breaks the filling of the current line and starts on the next line of output.

.rm - The right margin command commences adjusting of the right margin. It is enabled by default.

.mr - This is the reverse of the .rm command, and it ceases the adjusting of the right margins.

.ss - This command orders FMT to output with single spacing between lines. It is equivalent to .ls1. It is enabled by default.

.ds - Similar to .ss, this is the double space command. It is equivalent to .ls2.

.ts - This is the triple space command. It is equivalent to .ls3.

.lsX - This is the general line space command. X must a number from 1 to F hexadecimal.

.llXX - This sets the line length to XX, where XX is a hexadecimal number. Both digits must be given. The default is 65 decimal, 41 hexadecimal.

.plXX - This is the page length command. The argument is hexadecimal. The default is 88 decimal, 58 hexadecimal.

.inXX - The indent command sets the indentation. The default is 00.

.ceX - This is the center command. The next X lines are centered on the page.

.ulX - The underline command will underline the next X lines of the input text. Special characters will not be underlined (space, period, comma, etc.)

.he - This is the header command. The header will be placed on the third line of each page. Anything will be taken literally after the .he to be the text for the header. The header is not subject to filling, line length, or indentation. The special character '%' has a particular meaning to the header command. Whenever the percent is encountered in the header, it will be converted to the page number. The default is no header.