CGTM 71
JULY, 1969
JOHN R. EHRMAN

Features Added to the FORMAT Text-Processing Program

Features Added to the FORMAT Text-Processing Program

The FORMAT text-processing program (reference 1) provides a simple but powerful means of generating document-quality printed materials on a computer. Because FORMAT has proved to be of considerable help in a number of applications (references 2-5), user experience with it has shown several areas where small improvements could be made to either the input or the output facilities. Among these are:

1. Provision for dropping a character other than a period when tabbing with the 'D' command operand.

2. The ability to control the width of a control card, in the same way as can be done for text input records.

3. The ability to overprint pages more than one time, so as to eliminate much of the blur or grain that comes from the printer ribbon.

4. The ability to enter superscripts with the "ØN" convention even when the "SPECIAL KEYPUNCH" control card is in effect.

5. The ability to use another character in place of the "non-trivial blank" (0-2-8 punch) so that its use is possible from terminal devices.

6. Allowing the use of non-trivial blanks to center text and figures; currently, trailing non-trivial blanks are ignored for such purposes.

7. Backspacing and overprinting of individual characters.

8. Elimination of underlines under trailing punctuation.

All of the above facilities have been implemented locally, in a version of FORMAT known as FORMAT4E. The new control cards and a description of the use of the new facilities is given below. An important aspect of these new facilities is that they present no conflicts with present usage, since a correctly-prepared input deck for Level 4.0 of FORMAT will produce identical results under FORMAT4E (except for a small number of processor errors which have been corrected through the courtesy of the author of FORMAT).

New Control Cards

1. DROP CHARACTER FOR 'D' COMMAND IS xxx

Normally, the character dropped by the 'D' command operand will be a period. This control card may be used to change that character, as follows: if "xxx" is a number between 10 and 51,

then the drop character will be the corresponding special character; if "xxx" lies between 64 and 255, the drop character will be the EBCDIC character whose representation has that value; if it is zero or omitted, then the drop character will be reset to a period. The default character is a period, of course. As an example, the control card "DROP 241" would drop the digit "1" when the 'D' command operand is used.

2. DARK PRINT EACH PAGE n TIMES

Normally, each line on the output page will be printed once. If n has a value from 2 to 5, each line will be printed that many times, with the extra lines being printed on top of the previous one. This allows darker printing of the page. If n is 0 or 1, it is set to 1, and if it is greater than 5, it is set to 5. Note that the number of times each line is printed is determined by the value of n that is in effect at the time the page is printed, so that it is not possible to print portions of a page in "boldface". The default is single printing.

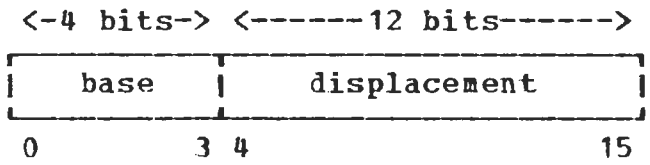
3. NULL CHARACTER SWITCH SET TO n

Non-trivial blanks (or null characters) are normally ignored for centering or underlining purposes when they are at the trailing end of a word. If n has the value 2, they will not be ignored when computing centering positions, and may be underlined. If n has any other value, it will be set to 1, which implies that null characters will be treated normally. The default setting is 1. The effect of this card does not apply in Titles or Footers.

4. BACKSPACE CHARACTER IS SPECIAL CHARACTER NUMBER nn

To simulate the action of the backspace key on a typewriter, one of the special characters may be designated as the "backspace" character, except for special characters numbered 43 (ø) and 51 (!). The action of this character is as follows: the character which follows the backspace character will be printed over the character preceding the backspace character, unless the following character after the backspace is a blank, in which case the backspace character prints normally. Multiple backspaces have no more effect than a a single one, in that they all cause only a single backspace, and the only character which will overprint the character preceding the first backspace will be the character following the last backspace. The number nn must lie between 10 and 50; if it does not, backspacing will be turned off and no character will be recognized as a backspace. Note that the backspace character may be entered in its actual (character) form or in its special (!nn) form. The default action is that no backspaces are recognized.

To give some examples: suppose the backspace character is number 50, the question mark (?). Then the input characters 0?- would produce θ, /?0 would produce Ø, and letǿt?_er would produce letTer. Note that special characters may be used for overprinting, so that =?| would produce ‡. The figure below makes use of backspacing to print the "corners" at the edges of the boxes.



The backspace character itself may not be used for overprinting. Backspacing does not apply in Titles and Footers.

At most 99 backspaces are allowed on a single page. Any backspaces following the 99th will be ignored, and the backspace character will print normally. An error message will flag the location of the 100th backspace on the page.

5. CONTROL CARDS END IN COLUMN nn

This control card allows the user to control the position of the right-hand margin of a control card in the same way as can be done for text input via the "CARD FIELD" control card. If the value of nn is less than 7 or greater than 80, it will be set to 80. This control card takes effect starting with the following control card, and does not affect cards with Title and Footer texts.

6. NONTRIVIAL BLANK IS REPRESENTED BY SPECIAL CHARACTER nn

In order to facilitate the use of the nontrivial blank from devices (such as terminals) which do not allow it to be entered in the source stream, the user may make the appearance of one of the special characters be equivalent to the presence of a nontrivial blank. The number nn must be between 10 and 51; otherwise no character will be replaced by the nontrivial blank when it is encountered. Note that the actual special character must be present to be replaced, and not the "representation" !nn, which will be treated normally. The default is that no such equivalence is made. This equivalence does not have any effect in Titles or Footers.

Modifications to Existing Features

7. SPECIAL KEYPUNCH IS A 2741

This small change was made to the "SPECIAL KEYPUNCH" control card, which will allow the user to perform the usual upper and lower case input from a terminal, but still be able to obtain superscripts through the use of the "ØN" technique. If there is any other numeric quantity on this control card, the normal "SPECIAL KEYPUNCH" will be assumed. Note that the effect of this card can be changed from 2741 mode to normal SPECIAL KEYPUNCH mode and back, but there is no way to return from either to the normal mode where Upper-case-only input is assumed.

8. Underlining

The underlining algorithm of FORMAT was modified so that the last character of an underlined string would not be underlined if it was one of the following ten punctuation or special characters: period, comma, colon, semicolon, question mark, exclamation point, apostrophe, quotation mark, apostrophe, and left or right parenthesis. In addition, the first character of the string is not underlined if it is a left parenthesis, a quotation mark, or an aopstrophe. If these characters must be underlined in the first or last positions, the backspace and overprint method should be used.

References

1. FORMAT, a Text-Processing Program, by Gerald M. Berns, available from IBM's Program Information Department as program 360D.06.0.003.

2. CIL, a Compiler Implementation Language, by David Gries, SLAC Report 102.

3. Disk Monitor for the SDS 9300, by A. E. Gromme, SLAC Computation Group Technical Memorandum No. 68.

4. Lecture Notes in Compiler-Writing for Computer Science 236, given at Stanford University during the Winter and Spring quarters of 1968-1969, written by David Gries.

5. Lecture Notes for Computer Science 139, written by John R. Ehrman.