

REFERENCE MANUAL

Q. E. D.

TIME-SHARING EDITOR

L. Peter Deutsch

Butler W. Lampson

University of California, Berkeley

Document No. 30.60.30

Issued May 7, 1965

Revised November 4, 1965

Contract SD-185

Office of Secretary of Defense

Advanced Research Projects Agency

Washington 25, D. C.

November 4, 1965

TABLE OF CONTENTS

|     |                                      |     |
|-----|--------------------------------------|-----|
| 1.0 | Introduction . . . . .               | 1-1 |
| 2.0 | Basic Features . . . . .             | 2-1 |
| 2.1 | Text Addressing . . . . .            | 2-1 |
| 2.2 | Text Input . . . . .                 | 2-2 |
| 3.0 | Line Manipulation Commands . . . . . | 3-1 |
| 4.0 | Line Edit Command . . . . .          | 4-1 |
| 5.0 | Other Commands . . . . .             | 5-1 |
| 6.0 | String Buffers . . . . .             | 6-1 |
| 7.0 | Panics From Q.E.D. . . . .           | 7-1 |

November 4, 1965

## 1.0 Introduction

Q. E. D. is a rather powerful program for editing symbolic text which runs under the 930 time-sharing system. Its input and output are symbolic files which can also be handled by the executive COPY command. It has extensive facilities for inserting, deleting and changing lines of text, a line edit feature, a powerful symbolic search feature, automatic tabs which may be set by the user, and ten string buffers. Text can be read from any file and written onto any file. A replace command permits all occurrences of a specified string of characters to be replaced with another string.

## 2.0 Basic Features

### 2.1 Text Addressing

The text being edited is held in a single buffer, called the main text buffer. It consists of a series of lines delimited by carriage returns (cr's). The line is the smallest addressable unit of text. In fact, it is the only addressable unit of text.

Lines may be addressed in the following ways:

- 1) By decimal numbers. The first line is numbered 1.
- 2) By . which refers to the current line. The value of . is changed by many of the editor's commands, as described below.

- 3) By \$, which refers to the last line.

- not / but :*
- 4) By labels. The structure `/text/` causes a search for the indicated text at the beginning of a line and followed by a character which is not a letter or digit. The text may include any characters except `/`. The search begins with the current line + 1 and cycles to the beginning of the buffer if it runs off the end. If it is successful, `.` addresses line found and the value of `/.../` is also this line. If no line with the specified label exists, QED prints ? and acts as though the erroneous search had never been requested. The search may be begun at line N by typing `N/text/` : for example, `./ABC/` begins the search at the current line.

- 5) By arbitrary text. The structure [text] causes a search for the indicated text anywhere in the buffer. The text may include any characters except ]. The search proceeds in the same way as the label search, and has the same effect when it succeeds or fails.
- 6) By a legal address followed by + or - followed by another legal address. The meaning is obvious. A negative result is converted to 1; one bigger than the number of lines in the buffer elicits ? from QED.

## 2.2 Text Input

There are several contexts in which QED expects text as input. This text is an arbitrary string of characters which is terminated by a D<sup>c</sup> (control D, or EOT). Carriage returns in the text will delimit lines when it is inserted into the main buffer, but have no significance as text terminators.

During text input corrections may be made with the following control characters, any of which may be iterated:

- A<sup>c</sup> deletes the preceding character and prints ↑
- W<sup>c</sup> deletes the preceding characters up to but not including the first preceding blank which is followed by a non-blank character. That is, all immediately preceding blanks are deleted, and then all characters up to the next preceding blank. The character \ is printed.
- Q<sup>c</sup> deletes the line being typed, or the preceding line if there is no text in the current line, and types ←. Repeated use of this character will delete several lines.

A control character is obtained by typing the character before the <sup>c</sup> with the CONTROL key held down. Characters may also be inserted into text from string buffers. Typing B<sup>c</sup>N, where N is a decimal digit, causes buffer N to be bodily inserted in the text. The B<sup>c</sup>N, of course, is not inserted.

The character V<sup>c</sup> causes the next character typed to be taken literally, regardless of any control function which it may normally serve.

[ = K<sup>s</sup>  
] = M<sup>s</sup>

2-3  
November 4, 1965

Pushing the rubout button during text input causes QED to ring the bell. No other action is taken unless the rubout button is pushed again with no intervening typing. In this case the text being entered is lost, the current operation is aborted and control returns to the QED command processor.

Any text input which is to be added to the main buffer (i.e., input to APPEND, INSERT or CHANGE) is terminated with carriage return regardless of whether a terminal carriage return has been typed by the user before the D<sup>c</sup>. No such action is taken for text input to searches or loaded into string buffers.

QED has tabs which can be set using the TAB command. They are initialized to 8, 16, 32 and 40. The character for tab is I<sup>c</sup>, which is so labeled on the teletype.

### 3.0 Line Manipulation Commands

In the following, A refers to any legal address. Spaces need not follow addresses. There are three basic line print commands:

- ~~er~~ f prints the next line
- ↑ prints the preceding line
- A/ prints the line addressed. The value of . is changed.
- A,A/ prints the block of lines between the two addressed, inclusive. The value of . is changed to the last line printed.
- @/ prints the entire main text buffer.

In addition to these, there are a number of commands specified by letters. Typing the first letter suffices to specify the command. If requested, QED will echo the rest of the command. All these commands must be terminated with a dot before they execute. The user is thus protected from inadvertently destroying a part of his text which he only wanted to print.

- A PRINT. prints the line addressed. The value of . is changed. PRINT is identical to : in its effects.
- A,A PRINT. prints the block of lines between the two addressed, inclusive. The value of . is changed to address the last line printed.
- PRINT. prints the entire main text buffer.
- APPEND. accepts text which it appends to the end of the buffer. Text is terminated with D<sup>c</sup> (control D, or EOT). Afterwards, . is moved to the last line of the old buffer.
- A INSERT. inserts text before line A and moves . to the last line inserted.
- A DELETE. deletes the line addressed and moves . to the previous line.
- A,A DELETE. deletes the block between the two lines addressed, inclusive, and moves . to the line before the first one deleted.

3-2  
July 20, 1965

A CHANGE. deletes the line addressed and inserts the text  
which follows the command. At the end of the  
insertion, . is moved to the last line inserted.

A,A CHANGE. does the obvious thing.

November 4, 1965

4.0 Line Edit Command

- A EDIT. causes the line addressed to be printed out. QED then goes into line edit mode, in which it accepts the following control characters:
- C<sup>c</sup> copies the next character of the original line to the edited line and types out that character.
- S<sup>c</sup> skips the next character of the original line and types out %.
- U<sup>c</sup> copies the original line up to the next tab stop.
- Z<sup>c</sup> C copies the original line up to the next occurrence of the character C. The character C is not echoed until its next occurrence is reached. Z<sup>c</sup> therefore produces output identical to that produced by multiple use of C<sup>c</sup>. If C does not appear in the line QED rings the bell and takes no other action.
- X<sup>c</sup> C is the same as Z<sup>c</sup> except that it deletes the characters passed over and types %. C is never typed.
- R<sup>c</sup> (retype). QED types line feed, then the rest of the original line, then on the next line the edited line so far. Editing may then be resumed. This control character is intended to permit recovery in cases where the user has become confused about the state of the edit.
- T<sup>c</sup> is the same as R<sup>c</sup> except that the new line is correctly aligned with the remainder of the old one. It takes longer.
- Y<sup>c</sup> copies the remainder of the old line to the new one without typing it. The new line is then typed out and becomes the old line for a continuation of the edit.

November 4, 1965

cr is added to the edited line and terminates the edit. The rest of the original line is ignored.

D<sup>C</sup>(EOT) causes the remainder of the original line to be typed out and copied to the edited line, which then replaces the original line.

F<sup>C</sup> is the same as D<sup>C</sup> except that the rest of the line is not typed.

text replaces corresponding text in the original line, character for character, unless in insert mode, when it is inserted without any characters of the original line being passed over. Characters typed when the original line is exhausted are always inserted on the end of the new line.

E<sup>C</sup> switches from replace to insert mode, and types a left angle bracket, or from insert back to replace mode, and types a right bracket. The mode is always replace at the beginning of a line edit.

The standard deletion characters, A<sup>C</sup>, W<sup>C</sup> and Q<sup>C</sup> may be used in a line edit.

The affect only that portion of the new line which has already been produced.

The command

A MODIFY. is identical to EDIT except that the line being edited is not typed out before the edit begins.

November 4, 1965

## 5.0 Other Commands

- READ FROM accepts a file name in the format required by the executive, reads the file and appends it to the main text buffer. Do not forget to clear the buffer first if it has contents which should not be retained.
- WRITE ON writes the main text buffer on the file specified after the command.
- A WRITE ON writes line A on the file specified after the command.
- A,B WRITE ON writes lines A through B on the file specified after the command.
- N KILL. kills the specified string buffer. To kill the main text buffer, use 1,\$DELETE.
- TABS. accepts a sequence of decimal numbers separated by commas and sets the tab stops at the specified positions. The string of numbers is terminated with dot.
- VERBOSE. causes QED to print the rest of each command after getting the first character. This is the normal mode.
- QUICK. suppresses printout of the remainder of each command.
- A= types out the address as an absolute line number
- A← types out the address as a line label, perhaps followed by + and a number.

## 6.0. String Buffers; Move and Substitute

Ten string buffers are available to the user. Their contents may be any string of text. Buffer 0 always contains the text most recently used in a search command. The other buffers may be loaded with

- N L~~OAD~~.                    which loads buffer N with the following text;
- A;N L~~OAD~~.                    which loads buffer N with the contents of line A;
- A,B;N L~~OAD~~.                    which loads buffer N with the contents of lines A through B.
- ;N L~~OAD~~                    . loads buffer N with the entire main text buffer

When a buffer is loaded it is cleared first.

A buffer may be printed with

- N BUFFER.                    which prints the contents of the specified buffer. N must be a digit between 0 and 9.

The contents of a buffer may be inserted into any text by typing  $B^C N$ , where N is a digit between 0 and 9.

A buffer may also be used as a source for commands in the same way. I.e.,  $B^C N$  is in all cases exactly equivalent to the string of characters in buffer N with the following exceptions:

- 1) An error in a command taken from a buffer causes the rest of the buffer to be ignored.
- 2) A buffer may not call another buffer. If  $B^C N$  is inserted into a buffer with  $V^C$ , it will cause a transfer to buffer N. In this case the characters following the N will never be accessed. Note that infinite loops can be constructed in this way.

The amount of space available for string buffers is limited. An attempt to load a buffer which would overflow this space will cause QED to type (FULL). The buffer will be cleared but not loaded.

## 7.0 Panics From Q.E.D.

The rubout button may be pushed at almost any time. It causes the current operation to be terminated. QED gives \* and awaits further commands. The value of . is not changed. If QED is in the middle of printing or writing a large amount of text, . will be set to the last line printed or written. As discussed above, rubout during text input rings the bell the first time and aborts the input only if it is pushed again with no intervening typing.

To return from QED to the exec, push rubout twice with no intervening typing.

If any QED operation causes a memory panic, the error message NO ROOM will be typed and the operation suspended. In the case of CHANGE and SUBSTITUTE, the deletion will be completed, but none of the insertion will be.

If QED executes an illegal instruction, it types FATAL ERROR and returns to the exec.

There are two more commands associated with string buffers.

A;nSUBSTITUTE types out WAIT? and expects YES or NO in response. If it gets YES, it searches the text, beginning on line A, for occurrences of the string in buffer 0. Each line with such an occurrence is typed out. The user may then type S or any other character. If S is typed the string in buffer n is substituted for the one found. In both cases, the scan then continues. When there are no occurrences of the string in buffer 0 in the entire text buffer the command is complete. If NO WAIT was specified, the process is the same except that no typing occurs and every substitution is made.

A,A;nGET loads buffer n with the specified lines, which are then deleted. It is useful for moving blocks of text.

0 QED FEATURES NEW IN 1.91 SYSTEM

0A JDH 23 OCT 67

1 IN RUNNING SYSTEM 1.91, WE HAVE HAD DIFFICULTIES BOTH BECAUSE I MADE A MISTAKE IN LOADING IT SO THE SCAN MECHANISM DIDN'T WORK RIGHT AND ALSO BECAUSE THERE HAVE BEEN CHANGES IN THE SYSTEM THAT AREN'T DOCUMENTED.

1A QED IS NOW WORKING PROPERLY.

1B THE CHANGES I KNOW ABOUT ARE LISTED BELOW.

2 THE SYMBOL "@" MEANS THE SAME THING AS "1,S".

3 IN THE "/", "PRINT", AND "SUBSTITUTE" COMMANDS, A NULL ARGUMENT LIST HAS A DIFFERENT MEANING FROM BEFORE.

3A FOR INSTANCE, FORMERLY "/" HAD THE SAME MEANING AS "1,S/".

3B NOW, "/" HAS THE SAME MEANING AS "./".

3B1 TO LIST THE ENTIRE TEXT BUFFER, TYPE "@/".

4 THE "SUBSTITUTE" COMMAND HAS BEEN CHANGED CONSIDERABLY.

4A FOR A "SUBSTITUTE" TO BE EFFECTIVE ON THE ENTIRE TEXT BUFFER, IT TOO SHOULD BE PRECEDED BY AN "@".

4B IN GIVING THE COMMAND, ONE MUST CHOOSE A DELIMITING CHARACTER WHICH DOESN'T OCCUR IN EITHER STRING INVOLVED IN THE SUBSTITUTION.

4B1 THE DELIMITER MUST NOT BE ":" SINCE THIS IS USED TO REQUEST A "WAIT" OR A LISTING DURING THE SUBSTITUTIONS.

4B2 TO REPLACE "SAM" WITH "BOB" THROUGHOUT THE BUFFER, TYPE "@SZBOBZSAMZ."

4B2A "Z" WAS USED AS THE DELIMITER.

4B2B THE COMMAND WOULD APPEAR ON THE TELETYPE AS:  
"\*@SUBSTITUTE ZBOBZ FOR ZSAMZ.

4C TO REQUEST A "WAIT" FOR EACH SUBSTITUTION, ADD A ":W" AFTER TYPING THE "S".

4C1 THE EXAMPLE ABOVE WITH A "WAIT" WOULD BE TYPED:  
"@S:WZBOBZSAMZ."

4C2 YOUR RESPONSES DURING THE EXECUTION OF THE SUBSTITUTE SHOULD BE EITHER "S" FOR YES OR "N" FOR NO. (I DON'T KNOW WHY)

4D TO REQUEST A LISTING OF ALL SUBSTITUTIONS WITHOUT A "WAIT", USE ":L" INSTEAD OF ":W".

5 THE "FINISHED" COMMAND HAS BEEN ADDED TO EXIT FROM QED  
BACK TO THE EXEC.

5A THIS COMMAND INCLUDES A FEATURE TO REMIND YOU IF YOU HAVEN'T  
WRITTEN OUT THE TEXT BUFFER.

?

\*