%MACRO UPONE FOR CAPITALIZING THE FIRST LETTER OF EACH WORD

by

Lawrence H. Muhlbaier, Duke University Medical Center
Judith A. Stafford, Duke University Medical Center

We have recently found the need to convert various files from the normal data processing format of all capital letters to a mixture of upper and lower case. UPONE is specifically oriented to converting names and addresses from all uppercase to uppercase for the first letter in each word. The %MACRO has provision for special cases (Mc, III, etc) and is designed to be easily extended to provide for other special cases particular to an application.

UPONE is written in function format and directly modifies the string variable that is its argument.

As more and more word processors and micro-computers are being connected to mainframe computers, there is more and more interest in converting the text data typically found on large mainframe data bases into a more "socially acceptable" form, e.g. with upper and lower case. We like to see it on a typed page. Most of the mainframe databases that we have had to convert have their text in the form of names and addresses. The %MACRO presented here handles them quite nicely. The argument to UPONE is presumed to be in all upper case before transformation; after transformation the argument has only the first letter of each word capitalized, with the exception of certain special cases.

In planning a macro of this sort, one of the major questions that we had to answer is what is a "word"? Since names and addresses follow slightly different rules from regular English text, we borrowed from several style manuals to operationally define a "word" as a character string of one or more alphanumeric characters whose first character follows a space (" "), a period ("."), a hyphen ("-"); a single quote (""'), or a left parenthesis ("("). A period immediately followed by a letter appears in name suffixes such as "M.D.", the hyphen allows for hyphenated place or person names, and the single quote is to handle names such as "O'Toole." This method of defining a word will not correctly capitalize street names with embedded letters (e.g. "1238 Main Street").

This %MACRO will only operate in the environment of a DATA program. The execution of the %MACRO is in two steps. In the first, a %MACRO UPILOWSET is called without arguments to define working variables and arrays. UPILOWSET is specified just once in the DATA program. The second %MACRO, UPONE, has one argument, the string to be converted into upper and lower case. The argument string variable is modified by the %MACRO.

UPONE can handle a number of special cases in names and addresses. The prefix "Mc" is followed by a capitalized letter. The suffixes "II," "III," and "IV" are all handled properly. The letter following the prefix "Mac" or "Van" is not converted because we found as many names whose next letter was not capitalized (e.g. "Mack") as was (e.g. "MacDonald"). We also judged that errors like "Macdonald" looked better than "Mac" and "Vandeke" looked better than "VanCe."

Bugs and possible improvements:

The program will not properly convert some job-title type lines (e.g. "Board of Directors" comes out "Board Of Directors"). The current post office usage of state abbreviations has two letter codes that are both capitals; this program will not recognize these as codes so using UPONE on city-state strings may yield undesirable results. Similarly for compass directions. This program uses the "brute force" approach to obtain the upper case letters and their corresponding lower case letters. We would like at some time to have it based on the internal coding sequence of the letters and thus only need to define "A" and "a."
Program Listing for %UPONE

%macro uplowset;
options dquote;
length ul-u26 11-126 $1.;
array upper ul-u26; 
array lower 11-126;
u1="A"; u2="B"; u3="C"; u4="D"; u5="E"; u6="F"; u7="G"; u8="H";
u9="I"; u10="J"; u11="K"; u12="L"; u13="M"; u14="N"; u15="O";
u16="P"; u17="Q"; u18="R"; u19="S"; u20="T"; u21="U"; u22="V";
u23="W"; u24="X"; u25="Y"; u26="Z";
l1="a"; l2="b"; l3="c"; l4="d"; l5="e"; l6="f"; l7="g"; l8="h";
l9="i"; l10="j"; l11="k"; l12="l"; l13="m"; l14="n"; l15="o";
l16="p"; l17="q"; l18="r"; l19="s"; l20="t"; l21="u"; l22="v";
l23="w"; l24="x"; l25="y"; l26="z";
%mend uplowset;

%macro UPONE(upstring);
  _l_=length(upstring); /* length of original string */
do _j_=2 to _l_; 
  chr=substr(upstring,_j_,-1); /* previous character */
  /* These 'if' statements represent exception conditions */
  if (chr=' ' then do; /* Capitalize if necessary */
    do over upper; /* Do over upper */
      if substr(upstring,_j_,1)=lower then 
        substr(upstring,_j_,1)=upper;
    /* End; */
    goto _ENDJ_;
  end;
  if (_j_=1) then if (substr(upstring,_j_-2,3)=" II") 
    do; /* Do over upper */
      if substr(upstring,_j_-2,3)=" IV" then goto _ENDJ_; 
      if (_j_-1=1) then 
        substr(upstring,_j_-1)=upper;
    /* End; */
    goto _ENDJ_;
  end;
  if (_j_-4=1) then 
    if substr(upstring,_j_-2,4)=" III" then do;
      if (_j_-1=1) then 
        substr(upstring,_j_-1)=upper;
    /* End; */
    goto _ENDJ_;
  end;
/* Do the upper to lower case conversion. */
do over upper;
  if substr(upstring,_j_,1)=upper then 
    substr(upstring,_j_,1)=lower;
end;
_ENDJ_; end;
%mend UPONE;

/* Example of UPONE usage*/
data;
  input string1 & $30.;
  %uplowset;
  put string1 @;
%upone(string1); put @40 string1;
cards;

PATRICK O'LEARY 
CHICAGO, IL
DOCTOR FEELGOOD, M.D. 
BEVERLY HILLS, CA
PHILLY PHANATIC
VETERANS STADIUM

Patrick O'Leary 
Chicago, IL
Doctor Feelgood, M.D. 
Beverly Hills, Ca
Philly Phanatic
Veterans Stadium