```c
#include <stdio.h>

#define MAX_STRING_LENGTH 512

#define SPACE ' '

#define FULLSTOP '.'
#define COMMA ','
#define EXCLAMATION '!'
#define QUESTION_MARK '?'

int main(void)
{
    //function prototype
    int MyStrlen(char[]);
    char MyToUpper(char);

    //variable declarations
    char chArray[MAX_STRING_LENGTH], chArray_CapitalizedFirstLetterOfEveryWord
      [MAX_STRING_LENGTH]; // A Character Array Is A String
    int iStringLength;
    int i, j;

    //code

    // *** STRING INPUT ***
    printf("\n\n");
    printf("Enter A String : \n\n");
    gets_s(chArray, MAX_STRING_LENGTH);

    iStringLength = MyStrlen(chArray);
    j = 0;
    for (i = 0; i < iStringLength; i++)
    {
        if (i == 0) //First Letter Of Any Sentence Must Be A CAPITAL LETTER
            chArray_CapitalizedFirstLetterOfEveryWord[j] = MyToUpper(chArray[i]);

        else if (chArray[i] == SPACE) //First Letter Of Every Word In The
          Sentence Must Be A CAPITAL LETTER. Words Are Separated By Spaces.
        {
            chArray_CapitalizedFirstLetterOfEveryWord[j] = chArray[i];
            chArray_CapitalizedFirstLetterOfEveryWord[j + 1] = MyToUpper(chArray
              [i + 1]);

            //SINCE, ALREADY TWO CHARACTERS (AT INDICES 'i' AND i + 1 HAVE BEEN
              CONSIDERED IN THIS else-if BLOCK...WE ARE EXTRA-INCREMENTING 'i'
              AND 'j' BY 1
            j++;
            i++;
        }

        else if ((chArray[i] == FULLSTOP || chArray[i] == COMMA || chArray[i] ==
```

```
            EXCLAMATION || chArray[i] == QUESTION_MARK) && (chArray[i] !=
            SPACE)) //First Letter Of Every Word After Punctuation Mark, In The
            Sentence Must Be A CAPITAL LETTER. Words Are Separated By Punctuations.
48          {
49              chArray_CapitalizedFirstLetterOfEveryWord[j] = chArray[i];
50              chArray_CapitalizedFirstLetterOfEveryWord[j + 1] = SPACE;
51              chArray_CapitalizedFirstLetterOfEveryWord[j + 2] = MyToUpper(chArray
                [i + 1]);
52
53              // SINCE, ALREADY TWO CHARACTERS (AT INDICES 'i' AND i + 1 HAVE BEEN
                CONSIDERED IN THIS else-if BLOCK...WE ARE EXTRA-INCREMENTING 'i' BY
                 1
54              // SINCE, ALREADY THREE CHARACTERS (AT INDICES 'j' AND (j + 1) AND (j
                 + 2) HAVE BEEN CONSIDERED IN THIS else-if BLOCK...WE ARE EXTRA-
                INCREMENTING 'j' BY 2
55              j = j + 2;
56              i++;
57          }
58
59          else
60              chArray_CapitalizedFirstLetterOfEveryWord[j] = chArray[i];
61
62          j++;
63      }
64
65      chArray_CapitalizedFirstLetterOfEveryWord[j] = '\0';
66
67          // *** STRING OUTPUT ***
68      printf("\n\n");
69      printf("String Entered By You Is : \n\n");
70      printf("%s\n", chArray);
71
72      printf("\n\n");
73      printf("String After Capitalizing First Letter Of Every Word : \n\n");
74      printf("%s\n", chArray_CapitalizedFirstLetterOfEveryWord);
75
76      return(0);
77  }
78
79  int MyStrlen(char str[])
80  {
81      //variable declarations
82      int j;
83      int string_length = 0;
84
85      //code
86      // *** DETERMINING EXACT LENGTH OF THE STRING, BY DETECTING THE FIRST
          OCCURENCE OF NULL-TERMINATING CHARACTER ( \0 ) ***
87      for (j = 0; j < MAX_STRING_LENGTH; j++)
88      {
89          if (str[j] == '\0')
90              break;
```

```
 91          else
 92              string_length++;
 93      }
 94      return(string_length);
 95  }
 96
 97  char MyToUpper(char ch)
 98  {
 99      //variable declaration
100      int num;
101      int c;
102
103      //code
104
105      // ASCII VALUE OF 'a' (97) - ASCII VALUE OF 'A' (65) = 32
106      // THIS SUBTRACTION WILL GIVE THE EXACT DIFFERENCE BETWEEN THE UPPER AND
           LOWER CASE COUNTERPARTS OF EACH LETTER OF THE ALPHABET
107      // IF THIS DIFFERENCE IS SUBTRACTED FROM THE ASCII VALUE OF A LOWER CASE
           LETTER, THE RESULTANT ASCII VALUE WILL BE THAT OF ITS UPPER CASE COUNTER-
           PART, HENCE, HELPING US TO FIND ITS UPPER CASE LETTER !!!
108      // ASCII VALUES OF 'a' to 'z' => 97 TO 122
109      // ASCII VALUES OF 'A' to 'Z' => 65 TO 90
110      num = 'a' - 'A';
111
112      if ((int)ch >= 97 && (int)ch <= 122)
113      {
114          c = (int)ch - num;
115          return((char)c);
116      }
117
118      else
119          return(ch);
120  }
121
```