INTRODUCTION TO PROGRAMMING LAB. Experiment 9: Array Processing

OBJECTIVES

1. Learn how to multiple pass arrays and pointers to functions.
2. Learn simple string functions in standard library

GENERAL INFORMATION

Standard c library functions include many NULL terminated character array handling functions. Most used functions are:

- `int strcmp(char *S1, const char *S2)`: compare S1 and S2 (return 0 if same).
- `char *strcpy(char *S1, const char *S2)`: copy S2 into S1.
- `char *strcat(char *S1, const char *S2)`: append S2 to S1.
- `size_t strlen(const char *S)`: determine the length of a string.
- `char *strncat(char *S1, char *S2, size_t n)`: append n characters from S2 to S1.
- `int strncmp(char *S1, char *S2, size_t n)`: compare first n characters of two strings.
- `char *strncpy(char *S1, const char *S2, size_t n)`: copy first n characters of S2 to S1

**Example:**

```c
strcpy(B, "Veli efendi"); strcat(A, B, 5); strcat(A, C);
puts(A); /* prints Ali Veli 49-50 */
if(strncmp(A, "49", 2) == 0) {
  printf("%s-%s\n", A, A+12);
}
```

EXPERIMENT

1. Write a program that asks the user to enter a string. The program shall print out the string with upper case initials and lower case the rest.
   
   Example screen; (user entered text is shown bold)

   
Enter string: osmangazi university MMF 1970
   Osmangazi University MMF 1970

2. Declare two parallel arrays for names and indexes as shown below.

   ```c
   char Name[100] = "ahmet mehmet ayse fatma"; /* filled for example */
   int Inx[50] = {};
   ```

   Write a function that accepts these arrays as arguments. In this function, using for loop, populate the Str array with indexes of the initial characters of the names separated with space characters. For the example arrays, Inx[] should be 0, 6, 13 and 18 respectively after the function call. Function returns the number of indexes found. The indexes should be printed out in function main().

   Example: (user entered string is shown bold)

   ```c
   Enter string: ali veli salami esra nuray
   Indexes: 0, 4, 9, 16, 21
   ```

3. Print out the names whose indexes are found above. Remember to replace space characters with NULL characters (easier).

   **QUESTIONS:**

   1. How would you handle character array pointers (character array arrays). Can you use this to separate input string into multiple arrays?