CSE 115 Lab on Strings and 1D Array

1. Passing array as an argument of a function:

```c
#include <stdio.h>
double getAverage(int arr [], int size){
    int i;
    double avg, sum=0;
    for (i=0; i< size; i++){
     sum += arr[i] ;
    }
    avg = sum/size;
    return avg;
}

void main()
{
  int balance [5] = {1000, 2, 3, 17, 50};
  double avg;
  avg = getAverage( balance , 5 );
  printf("Average is: %f ", avg);
}
```

Character:

ctype.h is a header file in C/C++ which contains functions for character handling ("ctype" stands for "**c**haracter **type**").

Some commonly used functions in ctype.h would make the uses of this header file clear:

isalnum()	It returns true value if the argument passed to it is an alphanumeric character.
isalpha()	It returns true value if the argument passed to it is an alphabet.
isdigit()	It returns true value if the argument passed to it is a decimal digit.
isspace()	It returns true value if the argument passed to it is a space.
isupper()	It returns true value if the argument passed to it is an uppercase alphabet.
islower()	It returns true value if the argument passed to it is a lowercase alphabet.
toupper()	It converts lowercase alphabet character to uppercase.
tolower()	It converts uppercase alphabet character to lowercase.

The prototype of all the above functions is similar:
int function_name (int argument) ;

String:

The **char *gets(char *s)** function reads a line from **stdin** into the buffer pointed to by **s** until either a terminating newline or EOF (End of File).

The **int puts(const char *s)** function writes the string 's' and 'a' trailing newline to **stdout**.

2. **Passing string as an argument of a function and update that string within the function:**

```c
#include <stdio.h>
#include<ctype.h>
#define SIZE 100

void changeCase(char s[])
{
    int i;

    for (i = 0; s[i] != NULL; i++)
    {
        if (isupper(s[i]))
            s[i] = tolower(s[i]);
        else if (islower(s[i]))
            s[i] = toupper(s[i]);
    }
}

void main ()
{
        char str[SIZE];

        printf ("Enter a string of length < %d : ", SIZE);
        gets(str);

        changeCase(str);
        puts(str);
}
```

Exercise:
1. **Write a C program that counts the number of vowels in an input string.**
2. **Write a C program that reads a string, create a new string containing all the characters the input string except the vowels in it, and then prints the new string.**
   Sample input/output:
   Enter a string: **Hello how are you?**
   Output string: Hll hw r y?
3. **Write a C program to print all unique letters in an input string.**
   Sample input/output:
   Enter a string: **Hello how are you?**
   Unique letters in the input string (ignoring differences between lowercase & uppercase letters):
   a, e, h, l, o, r, u, w, y
4. **Write a program that reads a string from user and then prints the number of times different letters appear in that string.**
   Sample input/output:
   Enter a string: **Hello how are you?**
   Frequencies of letters in the input string:
   a/A: 1, e/E: 2, h/H: 2, l/L: 2, o/O: 3, r/R: 1, u/U: 1, w/W: 1, y/Y: 1,

Assignment:

1. **Write a C program that deletes all duplicate letters from an input string.**
2. **Write a C program to put even and odd elements of an array in two separate arrays and show them.**
   Sample input/output:
   Input size of the array: **10**
   Input elements in array: **0 1 2 3 4 5 6 7 8 9**
   Even elements in array: 0 2 4 6 8
   Odd elements in array: 1 3 5 7 9
3. **Write a function that searches for a character in a string. The function should print true if found, otherwise print false.**

```
void search(char arr[], char key);
```

Sample Output 1:

```
Enter String: bangladesh
Search Key: g
Found
```

Sample Output 2:

```
Enter String: bangladesh
Search Key: v
Not found
```

4. **Write a function that returns the number of times a value appears in an array.**

```
int countSearchKey(int arr[], int size, int key);
```

Sample Output 1:

```
Enter array size: 5
Array Elements: 7 3 4 8 1
Search Key: 4
Search Key appears 1 times
```

Sample Output 2:

```
Enter array size: 7
Array Elements: 2 3 2 9 2 10 6
Search Key: 2
Search Key appears 3 times
```