Handling multiple Strings in C Language

1. Write a program to find the number of vowels in each of the 5 strings stored in two dimensional arrays, taken from the user.

```
#include <stdio.h>
int main()
{
    char a[5][20];
    int i, j, count;
    static int total = 0;
    printf("Enter 5 string\n");
    for (i = 0; i <= 4; i++)
    {
        fgets(a[i], 20, stdin);
    }
    for (i = 0; i <= 4; i++)
    {
        count = 0;
        for (j = 0; a[i][j]; j++)
        {
            if (a[i][j] == 'a' || a[i][j] == 'e' || a[i][j] == 'i' || a[i][j]
== 'o' || a[i][j] == 'u' || a[i][j] == 'A' || a[i][j] == 'E' || a[i][j] ==
'I' || a[i][j] == 'O' || a[i][j] == 'U')
            {
                count++;
                total++;
            }
        }
        printf("Number of vowels in %d string : %d\n", i + 1, count);
    }
     printf("Total number of vowels in all strings : %d\n", total);
    return 0;
}
```

```
Output:
Enter 5 string
Shekh Akhtar
Mukesh Rathore
Tarun Chandra
Gautam Sharma
Gajendra Yadav
Number of vowels in 1 string : 3
Number of vowels in 2 string : 5
Number of vowels in 3 string : 4
Number of vowels in 4 string : 5
Number of vowels in 5 string : 5
Total number of vowels in all strings : 22
```

2. Write a program to sort 10 city names stored in two dimensional arrays, taken from the user.

```
#include <stdio.h>
#include <string.h>
int main()
{
    char a[10][20], temp[20];
    int i, j, result;

    printf("Enter 10 city names\n");
    for (i = 0; i < 10; i++)
    {
        fgets(a[i], 20, stdin);
    }
```

```
    for (j = 1; j <= 9; j++)
    {
        for (i = 0; i <= 9 - j; i++)
        {
            result = strcmp(a[i], a[i+1]);
            if (result > 0)
            {
                strcpy(temp, a[i]);
                strcpy(a[i], a[i + 1]);
                strcpy(a[i + 1], temp);
            }
        }
    }

    printf("\nSorted city names\n");
    for (i = 0; i < 10; i++)
        printf("%s", a[i]);

    return 0;
}
```

```
Output:
Enter 10 city names
korba
raipur
bilaspur
bhilai
durg
raigarh
kota
dantewada
indore
bhopal

Sorted city names
bhilai
bhopal
bilaspur
dantewada
durg
indore
korba
kota
raigarh
raipur
```

3. Write a program to read and display a 2D array of strings in C language.

```
#include <stdio.h>
int main()
{
    char a[10][10];
    int n, i;

    printf("Enter number of strings to store in array (Max 10 string) : ");
    scanf("%d", &n);

    printf("Enter %d strings :\n", n);
    for (i = 0; i < n; i++)
    {
        fflush(stdin);
        fgets(a[i], 10, stdin);
    }
```

```
    printf("\nString stored in 2d array are :\n");
    for (i = 0; i < n; i++)
        printf("%s", a[i]);

    return 0;
}
```

```
Output:
Enter number of strings to store in array (Max 10 string) : 10
Enter 10 strings :
akhtar
mukesh
tarun
vijendra
gautam
ankit
gajendra
manish
ajay
shivam

String stored in 2d array are :
akhtar
mukesh
tarun
vijendra
gautam
ankit
gajendra
manish
ajay
shivam
```

4. Write a program to search a string in the list of strings.

```
#include <stdio.h>
#include <string.h>
int main()
{
    char a[10][10], c[10];
    int n, i, result;

    printf("Enter number of strings to store in array (Max 10 string) : ");
    scanf("%d", &n);

    printf("Enter %d strings :\n", n);
    for (i = 0; i < n; i++)
    {
        fflush(stdin);
        fgets(a[i], 10, stdin);
    }

    printf("Enter a string to search in the list : ");
    fgets(c, 10, stdin);

    for (i = 0; i < n; i++)
    {
        result = strcmp(c, a[i]);
        if (result == 0)
            break;
    }

    if (result == 0)
        printf("String found in the list of %d position", i + 1);
    else
```

```
        printf("Not found");

    return 0;
}
```

```
Output:
Enter number of strings to store in array (Max 10 string) : 6
Enter 6 strings :
delhi
mumbai
kolkata
hyderabad
bengaluru
bhopal
Enter a string to search in the list : bhopal
String found in the list of 6 position
```

5. Suppose we have a list of email addresses, check whether all email addresses have '@' in it. Print the odd email out.

```
#include<stdio.h>
int main()
{
    char a[20][20];
    int i, j, n;
     printf("Enter number of email addresses to store in array (Max 20 email
addresses) : ");
    scanf("%d", &n);

    printf("Enter %d email addresses :\n", n);
    for (i = 0; i < n; i++)
    {
        fflush(stdin);
        fgets(a[i], 20, stdin);
    }

    printf("\nThe odd email addresses: \n");
    for (i = 0; i < n; i++)
    {
        for (j = 0; a[i][j] != '\0'; j++)
        {
            if (a[i][j] == '@')
                break;
        }
        if (a[i][j] == '\0')
        {
            printf("%s", a[i]);
        }
    }

    return 0;
}
```

```
Output:
Enter number of email addresses to store in array (Max 20 email addresses) :
5
Enter 5 email addresses :
shekh.akhtar@outlook.com
mgajendra2654.com
kamrun@gmail.com
vinni4556.yahoo.com
kj41525@gmail.com

The odd email addresses:
```

```
mgajendra2654.com
vinni4556.yahoo.com
```

6. Write a program to print the strings which are palindrome in the list of strings.

```
#include<stdio.h>
#include<string.h>
int main()
{
    char a[10][20], temp[20];
    int i, n;

    printf("Enter how many strings do you want to input : ");
    scanf("%d", &n);

    printf("Enter %d string : \n", n);
    for (i = 0; i < n; i++)
    {
        fflush(stdin);
        gets(a[i]);
    }

    printf("Palindrome in the list of strings are as follows : \n");
    for (i = 0; i < n; i++)
    {
        strcpy(temp, a[i]);
        strrev(temp);
        int r = strcmp(temp, a[i]);
        if (r == 0)
        {
             printf("%s", a[i]);
             printf("\n");
        }
    }

    return 0;
}
================================================================================
Output:
Enter how many strings do you want to input : 6
Enter 6 string :
akhtar
madam
naman
level
manish
radar
Palindrome in the list of strings are as follows :
madam
naman
level
radar
```

7. From the list of IP addresses, check whether all ip addresses are valid.

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main()
{
    char a[10][20], b[10][20], *ptr = NULL;
    int i, j = 0, n, count = 0, flag = 0;
     printf("Enter a number, how much ip addresses you will be enter (Max 10
ip addresses): ");
    scanf("%d", &n);
```

```
    printf("Enter %d IP addresses\n", n);
    for (i = 0; i < n; i++)
    {
        fflush(stdin);
        gets(a[i]);
        strcpy(b[i], a[i]);
    }
    printf("\nInvalid IP addresses are as follows :\n");
    while (j < n)
    {
        ptr = strtok(a[j], ".");
        while (ptr != NULL)
        {
            count++;
            int x = atoi(ptr);
            if (x < 0 || x > 255)
            {
                printf("%s\n", b[j], i);
                flag = 1;
                break;
            }
            ptr = strtok(NULL, ".");
        }

        if ((count < 4 || count > 4) && flag != 1)
        {
            printf("%s\n", b[j]);
        }
        count = 0;
        flag = 0;
        j++;
    }
    return 0;
}
```

```
================================================================================
Output:
Enter a number, how much ip addresses you will be enter (Max 10 ip
addresses): 5
Enter 5 IP addresses
192.168.25.2
198.168.300.2
198.165.2.3
198.356
198.12.3.3.3

Invalid IP addresses are as follows :
198.168.300.2
198.356
198.12.3.3.3
```

8. Given a list of words followed by two words, the task is to find the minimum distance between the given two words in the list of words.
(Example : s = {“the”,”quick”,”brown”,”fox”,”quick”}
word1 = “the”, word2 = “fox”, OUTPUT : 2)

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main()
{
    char a[10][20], word1[20], word2[20];
    int i, n, w1 = -1, w2 = -1, min = 10000, temp;
```

```
    printf("Enter a number, how much words you will be enter (Max 10 words):
");
    scanf("%d", &n);

    printf("Enter %d words\n", n);
    for (i = 0; i < n; i++)
    {
        fflush(stdin);
        gets(a[i]);
    }

    printf("Enter 2 words in the above list to find minimum distance between
them\n");
    gets(word1);
    gets(word2);

    for (i = 0; i < n; i++)
    {
        if (strcmp(a[i], word1) == 0)
            w1 = i;
        if (strcmp(a[i], word2) == 0)
            w2 = i;

        if (w1 != -1 && w2 != -1)
        {
            temp = abs(w2 - w1);
            if (temp < min)
                min = temp;
        }
    }
    printf("\nThe minimum distance between the given two words is %d", min -
1);
    return 0;
}
```

```
================================================================================
Output:
Enter a number, how much words you will be enter (Max 10 words): 10
Enter 10 words
bhopal
indore
bengaluru
mumbai
pune
hyderabad
delhi
gujrat
kolkata
raipur

Enter 2 words in the above list to find minimum distance between them
kolkata
bengaluru

The minimum distance between the given two words is 5
```

9. Write a program that asks the user to enter a username. If the username entered is one of the names in the list then the user is allowed to calculate the factorial of a number. Otherwise, an error message is displayed

```
#include <stdio.h>
#include <string.h>
int factorial(int);
int main()
{
```

```
    char username[20], a[5][20] = {"akhtar", "mukesh", "gautam", "vijendra",
"tarun"};
    int i, n, flag = 0;

    printf("Enter username : ");
    fflush(stdin);
    gets(username);

    for (i = 0; i < 4; i++)
    {
        if (strcmp(username, a[i]) == 0)
        {
            printf("Username is available\n");
            printf("Enter a number to calculate factorial : ");
            scanf("%d", &n);
            printf("Factorial of a number %d is %d", n, factorial(n));
            flag = 1;
        }
    }

    if (flag == 0)
        printf("\nUsername or Password not matched");

    return 0;
}

int factorial(int a)
{
    int i, fact = 1;
    for (i = 1; i <= a; i++)
        fact = fact * i;
    return (fact);
}
================================================================================
Output:
Enter username : gautam
Username is available
Enter a number to calculate factorial : 5
Factorial of a number 5 is 120
```

10. Create an authentication system. It should be menu driven.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main()
{
    char username[20], password[20], a[4][2][20] = {{"akhtar", "22121995"},
{"mukesh", "12345678"}, {"gautam", "16101996"}, {"vijendra", "19121995"}};
    int i, n;

    while (1)
    {
        int flag = 0;
        printf("\n1. To login\n");
        printf("2. For exit\n");
        printf("Enter your choice : ");
        scanf("%d", &n);
        switch (n)
        {
        case 1:
            printf("Enter username : ");
            fflush(stdin);
            gets(username);
            printf("Enter password : ");
```

```
            fflush(stdin);
            gets(password);

            for (i = 0; i < 4; i++)
            {
                    if (strcmp(username, a[i][0]) == 0 && strcmp(password,
a[i][1]) == 0)
                {
                    printf("Successful login\n");
                    flag = 1;
                }
            }

            if (flag == 0)
                printf("\nUsername or Password not matched\n");
            break;

        case 2:
            exit(0);
        }
    }

    return 0;
}
```

==========

```
Output:

1. To login
2. For exit
Enter your choice : 1
Enter username : akhtar
Enter password : 22121995
Successful login

1. To login
2. For exit
Enter your choice : 1
Enter username : akhtar
Enter password : 123456

Username or Password not matched

1. To login
2. For exit
Enter your choice : 2
```