Assignment - 19 A Job Ready Bootcamp in C++, DSA and IOT MySirG

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 str[5][100];

for (int i = 0; i < 5; i++)

fgets(str[i], 100, stdin);

int count;

for (int i = 0; i < 5; i++)

{

count = 0;

for (int j = 0; str[i][j]; j++)

{

if (str[i][j] == 'a' || str[i][j] == 'e' || str[i][j] == 'i' || str[i][j] == 'o' || str[i][j] == 'u' || str[i][j] == 'A' || str[i][j] == 'E' || str[i][j] == 'I' || str[i][j] == 'O' || str[i][j] == 'U')

count++;

}

printf("\n%d String contan %d Vowel", i + 1, count);

}

return 0;

}

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

the user.

#include <stdio.h>

void strSort(char str[])

{

for (int i = 0; str[i]; i++)

{

for (int j = i + 1; str[j]; j++)

{

if (str[i] > str[j])

{

char ch = str[i];

str[i] = str[j];

str[j] = ch;

}

}

}

}

int main()

{

char str[10][100];

for (int i = 0; i < 10; i++)

fgets(str[i], 100, stdin);

printf("Sorted string: \n");

for (int i = 0; i < 10; i++)

{

strSort(str[i]);

printf("%s", str[i]);

}

return 0;

}

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

#include <stdio.h>

int main()

{

int noOfString;

printf("How many string u want to enter: ");

scanf("%d", &noOfString);

printf("Enter %d String: ", noOfString);

fflush(stdin); // for clear buffer

char str[noOfString][100];

for (int i = 0; i < noOfString; i++)

fgets(str[i], 100, stdin);

printf("Strings: \n");

for (int i = 0; i < noOfString; i++)

{

printf("%d String - %s", i + 1, str[i]);

}

return 0;

}

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

#include <stdio.h>

#include <string.h>

int main()

{

int noOfString;

printf("How many string u want to enter: ");

scanf("%d", &noOfString);

printf("Enter %d String: ", noOfString);

fflush(stdin); // for clear buffer

char str[noOfString][100];

for (int i = 0; i < noOfString; i++)

fgets(str[i], 100, stdin);

char search[100];

printf("\nEnter Search string: ");

fgets(search, 100, stdin);

int flag = 0;

for (int i = 0; i < noOfString; i++)

{

if (strcmp(str[i], search) == 0)

{

flag = 1;

break;

}

}

if (flag)

printf("Yes!, we got'it string");

else

printf("No!, we don't got'it");

return 0;

}

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>

#include <string.h>

int main()

{

int noOfString;

printf("How many string u want to enter: ");

scanf("%d", &noOfString);

printf("Enter %d String: ", noOfString);

fflush(stdin); // for clear buffer

char str[noOfString][100];

for (int i = 0; i < noOfString; i++)

fgets(str[i], 100, stdin);

printf("\nnot @ contain Email\n\n");

for (int i = 0; i < noOfString; i++)

{

int flag = 0;

for (int j = 0; str[i][j]; j++)

{

if (str[i][j] == '@')

{

flag = 1;

break;

}

}

if (flag == 0)

printf("%s", str[i]);

;

}

return 0;

}

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

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 ip[] = "23.100.453.255";

int countDot = 0, countVal = 0;

char \*a = strtok(ip, ".");

while (a != NULL)

{

int x = atoi(a);

if (x >= 0 && x <= 255)

countVal++;

a = strtok(NULL, ".");

countDot++;

}

if (countDot == 4 && countVal == 4)

printf("Valid ip address!");

else

printf("Invalid ip addresss!");

return 0;

}

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.

**#include <stdio.h>**

**#include <string.h>**

**int main()**

**{**

**char words[5][20] = {"My", "Name", "Is", "Sachin", "Payasi"};**

**char word1[] = "My";**

**char word2[] = "Sachin";**

**int i, indWord1 = -1, indWord2 = -1, min = 100000;**

**for (i = 0; i < 5; i++)**

**{**

**if (strcmp(words[i], word1) == 0)**

**indWord1 = i;**

**if (strcmp(words[i], word2) == 0)**

**indWord2 = i;**

**if (indWord1 != -1 && indWord2 != -1)**

**min = abs(indWord1 - indWord2);**

**}**

**printf("Minimum distance of two words is: %d", min-1);**

**return 0;**

**}**

(Example : s = {“the”,”quick”,”brown”,”fox”,”quick”}

word1 = “the”, word2 = “fox”, OUTPUT : 1 )

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>

void factorial()

{

int num;

printf("Enter a number: ");

scanf("%d", &num);

int fact = 1;

for (int i = 1; i <= num; i++)

fact \*= i;

printf("Factorial is: %d", fact);

}

int main()

{

int noOfString;

printf("How many string u want to enter: ");

scanf("%d", &noOfString);

printf("Enter %d String: ", noOfString);

fflush(stdin); // for clear buffer

char str[noOfString][100];

for (int i = 0; i < noOfString; i++)

fgets(str[i], 100, stdin);

char user[100];

printf("\nEnter Username- ");

fgets(user, 100, stdin);

int flag = 0;

for (int i = 0; i < noOfString; i++)

{

if (strcmp(str[i], user) == 0)

{

flag = 1;

break;

}

}

if (flag)

factorial();

else

printf("Error!");

return 0;

}

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

#include <stdio.h>

#include <string.h>

int main()

{

int flag = 0, i;

char a[3][2][20] = {

{"Ajay", "123"},

{"Soni", "456"},

{"Silent", "123"}};

char userName[] = "Silent";

char passWord[] = "456";

for (i = 0; i < 3; i++)

{

if (strcmp(userName, a[i][0]) == 0 && strcmp(passWord, a[i][1])==0)

{

flag = 1;

break;

}

}

if (flag == 0)

printf("Username or Password doesn't match!");

else

printf("Login Successfull!");

return 0;

}