**Assignment-19 : Solution Name: Om Pant**

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.

Sol-

// 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>

#include<string.h>

int main(){

    int i,j,vowels=0;

    char names[5][25];

    printf("Enter 5 names\n");

    for(i=0; i<5;i++){

        gets(names[i]);

    }

    for(i=0;i<5;i++){

        for(j=0; names[i][j] != '\0';j++){

            if(names[i][j] == 'A' || names[i][j] == 'a'){

                vowels++;

            }

            else if(names[i][j] == 'E' || names[i][j] == 'e'){

                vowels++;

            }

            else if(names[i][j] == 'I' || names[i][j] == 'i'){

                vowels++;

            }

            else if(names[i][j] == 'O' || names[i][j] == 'o'){

                vowels++;

            }

            else if(names[i][j] == 'U' || names[i][j] == 'u'){

                vowels++;

            }

        }

    }

    printf("Number of Vowels: %d\n",vowels);

    return 0;

}

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

Sol-

// 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(){

    int i,j,vowels=0;

    char cities[10][25];

    char temp[25];

    printf("Enter 10 city names\n");

    for(i=0; i<10;i++){

        gets(cities[i]);

    }

    //sorting the array

    for(i=0;i<9;i++){

        for(j=i;j<10;j++){

            if(strcmp(cities[i], cities[j]) >0){

                strcpy(temp,cities[i]);

                strcpy(cities[i],cities[j]);

                strcpy(cities[j],temp);

            }

        }

    }

    //printing sorted array

    printf("In Sorted Order:\n");

    for(i=0;i<9;i++){

        printf(" %s ",cities[i]);

    }

    return 0;

}

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

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

#include<stdio.h>

#include<string.h>

int main(){

    int i,j;

    char strings[5][25];

    //reading strings

    printf("Enter 5 Strings\n");

    for(i=0; i<5;i++){

        gets(strings[i]);

    }

    //Display strings

    printf("Strings in 2D Arrays are:\n");

    for(i=0;i<5;i++){

        printf(" %s ",strings[i]);

    }

    return 0;

}

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

Soln-

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

#include<stdio.h>

#include<string.h>

int main(){

    int flag = 0;

    char strings[5][25] = {"akshay","Satyam","Vijayraj","robert","andrew"};

    char serachStr[25];

    printf("Enter string to search\n");

    gets(serachStr);

    for(int i=0; i<5;i++){

        if(strcmp(serachStr,strings[i]) == 0){

            flag = 1;

        }

    }

    if(flag){

        printf("String Found in List\n");

    }else{

        printf("String not Found\n");

    }

    return 0;

}

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

Sol-

// 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(){

    char emailList[][50] = {"abc@gamil.com","xyz@ad.com","hgsy.abc","pqrs@tq", "abcd@yx.edu","your@name.com", "po@drsingh1","xys#23.com"};

    int i;

    printf("Odd Emails from the List are:\n\n");

    for(i = 0;i<8;i++){

        if( strchr(emailList[i],'@') && strchr(emailList[i], '.')){

            //do nothing

        }

        else{

            printf("%s \n",emailList[i]);

        }

    }

    return 0;

}

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

Sol-

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

#include<stdio.h>

#include<string.h>

int Palindrom(char string[]){

    int i,j,flag=1;

    j = strlen(string) -1;

    for(i=0; i<=j; i++){

        if(string[i] == string[j]){

            j--;

        }

        else{

            flag = 0;

            break;

        }

    }

    if(j<i && flag != 0){

        return 1;

    }

    else{

        return 0;

    }

}

int main(){

    char strings[10][50] = {"hello","naman","yash","ajay","pradhan","radar", "level","madam","abccba","xyzyx"};

    int i,j;

    //to check palindroms in list

    printf("Palindrom strings in list are:\n");

    for(i=0;i<10;i++){

        if(Palindrom(strings[i])){

            printf("%s \n",strings[i]);

        }

    }

   return 0;

}

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

Sol-

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

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <ctype.h>

int validate\_number(char \*str) {

   while (\*str) {

      if(!isdigit(\*str)){

         return 0;

      }

      str++;

   }

   return 1;

}

int validate\_ip(char \*ip) {

   int i, num, dots = 0;

   char \*ptr;

   if (ip == NULL)

      return 0;

      ptr = strtok(ip, ".");

      if (ptr == NULL)

         return 0;

   while (ptr) {

      if (!validate\_number(ptr))

         return 0;

         num = atoi(ptr);

         if (num >= 0 && num <= 255) {

            ptr = strtok(NULL, ".");

            if (ptr != NULL)

               dots++;

         } else

            return 0;

    }

    if (dots != 3)

       return 0;

      return 1;

}

int main() {

    char ipAdd[][20] = {"192.168.4.1",

                        "172.16.253.1",

                        "192.800.100.1",

                        "125.512.100.abc"};

    for(int i=0;i<4;i++)

         validate\_ip(ipAdd[i])? printf("Valid \n"): printf("Not valid \n");

}

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 )

sol-

// 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<stdlib.h>

#include<string.h>

int shortestDistance(char list[][20], char w1[], char w2[]){

    int d1 = -1, d2 = -1;

    int ans = 99999;

    for(int i=0;i<5;i++){

        if(strcmp(list[i],w1) == 0)

            d1 = i;

        if(strcmp(list[i],w2) == 0)

            d2 = i;

        if(d1 != -1 && d2 != -1)

            ans = (ans < abs(d1 - d2))? ans: abs(d1 - d2);

    }

    return ans;

}

int main(){

    char list[][20] =  {"the","quick","brown","fox","quick"};

    char word1[20],word2[20];

    printf("Enter two words from list to find distance\n");

    gets(word1);

    gets(word2);

    printf("Distance: %d",shortestDistance(list,word1,word2));

    return 0;

}

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

Sol-

// 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,fact=1;

    printf("Enter a number to calculate factorial\n");

    scanf("%d",&num);

    if(num ==0 || num ==1){

        printf("Factorial : 1\n");

    }

    else{

        for(int i=1;i<=num;i++){

            fact = fact\*i;

        }

        printf("Factorial : %d\n",fact);

    }

}

int main(){

    char users[10][20] = {"ajay","prateek","prashant","aakash","vijay","raj","mihir", "om", "emanual","harsh"};

    char user[20];

    int flag=0;

    printf("Enter Username\n");

    gets(user);

    for(int i=0;i<10;i++){

        if(strcmp(user,users[i]) ==0 ){

            flag = 1;

        }

    }

    if(flag){

        factorial();

    }else{

        printf("User Not Found !!\n");

    }

    return 0;

}

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

Sol-

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

#include<stdio.h>

#include<string.h>

 int main(){

    char users[6][20] = {"ajay","aditya","harsh","himanshu","naveen","shekhar"};

    char passwords[6][50] = {"pass@123","123456","abcdef","911","naveen@12","sh#12"};

    char username[20];

    char password[50];

    printf("Enter Username\n");

    fgets(username,20,stdin);

    username[strlen(username)-1] = '\0';

    //AUTHENTICATION SYSTEM

    /\* if wrong username entered programm will exit. else checks password for corresponding user\*/

    for(int i=0;i<6;i++){

        if(strcmp(username,users[i]) == 0){

            printf("Enter Your Password\n");

            fgets(password,50,stdin);

            password[strlen(password)-1] = '\0';

            if(strcmp(password,passwords[i]) ==0){

                printf("Login Successfully\n");

            }

            else{

                printf("Wrong Password Entered\n");

            }

        }

    }

    return 0;

 }