Q1) Sum of prime numbers in given range.

#include <stdio.h>

int main()

{

    int lowerLimit, upperLimit, sum = 0,i;

    printf("Enter lower limit of range greater than 0\n");

    scanf("%d", &lowerLimit);

    while(lowerLimit<0){

        printf("Enter lower limit of range greater than 0\n");

        scanf("%d", &lowerLimit);

    }

    printf("Enter upper limit of range greater than 0\n");

    scanf("%d", &upperLimit);

    while(lowerLimit<0){

        printf("Enter lower limit of range greater than 0\n");

        scanf("%d", &lowerLimit);

    }

    for (i = lowerLimit; i <= upperLimit; i++)

    {

        if(i==0 || i==1) continue;

        int isPrime = 1;

        for (int j = 2; j\*j <= i; j++)

        {

            if (i % j == 0)

            {

                isPrime = 0;

                break;

            }

        }

        if(isPrime)

        {

            printf(" %d +", i);

            sum += i;

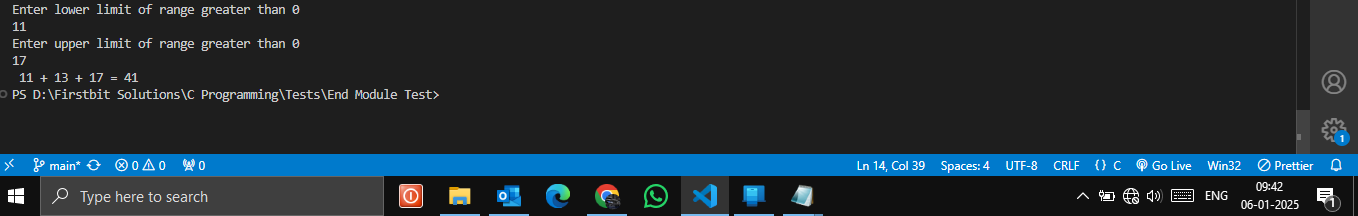
        }

    }

    printf("\b= %d\n", sum);

    return 0;

}



Q2) Replace any char from string with special char.

#include<stdio.h>

#include<string.h>

int main(){

    char str[50], ch, sc;

    printf("Enter string\n");

    gets(str);

    printf("Enter char you want to replace\n");

    scanf("%c", &ch);

    fflush(stdin);

    printf("Enter char you with which you to replace\n");

    scanf("%c", &sc);

    int len = strlen(str);

    printf("---Before---\n%s\n\n", str);

    fflush(stdin);

    for (int i = 0; i < len; i++)

    {

        if(str[i]==ch){

            str[i] = sc;

        }

    }

    printf("---After---\n%s\n\n", str);

    return 0;

}



Q3)Todo list.

#include <stdio.h>

#include <string.h>

typedef struct Todo

{

    int taskId;

    char desc[100];

    char status[10];

} Todo;

void displayAll(Todo \*trr, int \*index)

{

    for (int i = 0; i < \*(index); i++)

    {

        printf("Task Id --> %d\n", trr[i].taskId);

        printf("Description --> %s\n", trr[i].desc);

        printf("Status --> %s\n\n", trr[i].status);

    }

}

int addTask(Todo \*trr, int \*index)

{

    printf("Enter TaskId\n");

    scanf("%d", &trr[\*index].taskId);

    printf("Enter Description\n");

    scanf("%s", trr[\*index].desc);

    fflush(stdin);

    printf("Enter Status\n");

    scanf("%s", trr[\*index].status);

    fflush(stdin);

    (\*index)++;

    return 1;

}

int updateTask(Todo \*trr, int \*index)

{

    int tempId, choice, acIndex;

    printf("Enter Taskid u want to update\n");

    scanf("%d", &tempId);

    for (int i = 0; i < \*index; i++)

    {

        if(trr[i].taskId==tempId){

            acIndex = i;

        }

    }

    printf("1. Update Description\n");

    printf("2. Update Staus\n");

    printf("3. Exit\n");

    scanf("%d", &choice);

    switch (choice)

    {

    case 1:

    {

        char tempName[30];

        printf("Enter new name\n");

        scanf("%s", tempName);

        strcpy(trr[acIndex].desc, tempName);

        printf("Description updated successfully\n");

        break;

    }

    case 2:

    {

        char tempName[30];

        printf("Enter new status\n");

        scanf("%s", tempName);

        strcpy(trr[acIndex].status, tempName);

        printf("Staus updated successfully\n");

        break;

    }

    case 3: break;

    default:

        break;

    }

    return 1;

}

int main()

{

    Todo trr[10];

    int choice, index = 3;

    trr[0].taskId = 1;

    strcpy(trr[0].desc, "Grab Milk from market");

    strcpy(trr[0].status, "Pending");

    trr[1].taskId = 2;

    strcpy(trr[1].desc, "Take note of c from chatgpt");

    strcpy(trr[1].status, "Pending");

    trr[2].taskId = 3;

    strcpy(trr[2].desc, "Hello world");

    strcpy(trr[2].status, "Pending");

    while (1)

    {

        printf("1. Add Task\n");

        printf("2. Display Tasks\n");

        printf("3. Update Tasks\n");

        printf("0. Exit\n");

        scanf("%d", &choice);

        switch (choice)

        {

        case 1:

        {

            int res = addTask(trr, &index);

            res ? printf("Added successfully\n") : printf("error\n");

            break;

        }

        case 2:

        {

            displayAll(trr, &index);

            break;

        }

        case 3:

        {

            updateTask(trr, &index);

            break;

        }

        case 0:

        {

            break;

        }

        default:

            printf("Invalid choice");

            break;

        }

        if (choice == 0)

        {

            break;

        }

    }

    return 0;

}

