***Assignment 14 || Structure***

*Arjun Patel – FRN-13J1124/006*

Q1) Create a structure Book with data members as name, id, author, price. Accept the values of all these members from user and display them.

#include<stdio.h>

#include<string.h>

typedef struct Book{

    char name[30];

    int id;

    char author[30];

    double price;

} Book;

void fgetsInput(char\* str, int size){

    int len = strlen(str);

    fflush(stdin);

    if(fgets(str, size, stdin)){

        if(len>0 && str[len-1]=='\n') str[len-1] = '\0';

    }

    fflush(stdin);

}

void displayAll(Book\* books,int n){

    for (int i = 0; i < n; i++)

    {

        printf("Book Name -> %s", books[i].name);

        printf("Book id -> %d\n", books[i].id);

        printf("Author Name -> %s", books[i].author);

        printf("Book id -> %.2lf\n", books[i].price);

    }

}

int main(){

    Book books[3];

    for (int i = 0; i < 3; i++)

    {

        printf("---Book %d---\n", i+1);

        printf("Enter name of the book\n");

        fgetsInput(books[i].name, sizeof(books[i].name));

        printf("Enter id of book\n");

        scanf("%d", &books[i].id);

        printf("Enter name of the author\n");

        fgetsInput(books[i].author, sizeof(books[i].name));

        printf("Enter price of book\n");

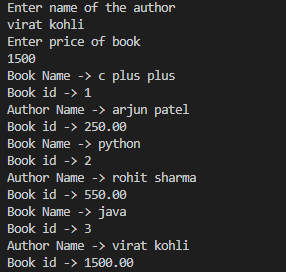
        scanf("%lf", &books[i].price);

    }

    displayAll(books, 3);

    return 0;

}



Q2)

#include<stdio.h>

typedef struct Time{

    int hr;

    int min;

    char sec;

} Time;

void displayAll(Time\* t,int n){

    for (int i = 0; i < n; i++)

    {

        printf("%d : %d : %d\n", t[i].hr, t[i].min, t[i].sec);

    }

}

int convertIntoSecs(Time t){

    int total = (t.hr\*3600) + (t.min\*60) + (t.sec);

    return total;

}

void add(Time t1, Time t2){

    int totalSecs = convertIntoSecs(t1) + convertIntoSecs(t2);

    int totalHrs =  totalSecs/3600;

    int hrRem = totalSecs%3600;

    int totalMin =  hrRem/60;

    int minRem = hrRem%60;

    int totalSec =  minRem%60;

    printf("%d : %d : %d\n", totalHrs, totalMin, totalSec);

}

int main(){

    Time t[2];

    for (int i = 0; i < 2; i++)

    {

        printf("---Time %d---\n", i+1);

        printf("Enter hrs\n");

        scanf("%d", &t[i].hr);

        printf("Enter minutes\n");

        scanf("%d", &t[i].min);

        printf("Enter secs\n");

        scanf("%d", &t[i].sec);

    }

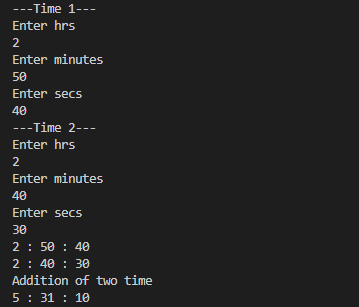
    displayAll(t, 2);

    printf("Addition of two time\n");

    add(t[0], t[1]);

    return 0;

}



Q3) Write a program to create an array for 10 players. For each player store name, no. of matches played, runs, wickets takes.

a. Create function to Accept the information of each player.

b. Create function to display the information of all the players

c. Display the information of player who made maximum runs and the one who took maximum number of wickets.

#include <stdio.h>

#include <string.h>

/\*

Write a program to create an array for 10 players. For each player store name, no. of matches played, runs, wickets takes.

a. Create function to Accept the information of each player.

b. Create function to display the information of all the players

c. Display the information of player who made maximum runs and the one who took maximum number of wickets.

\*/

typedef struct Players

{

    char name[30];

    int noOfMatches;

    int runs;

    int wickets;

} Players;

void fgetsInput(char \*str, size\_t size)

{

    fflush(stdin);

    if (fgets(str, size, stdin))

    {

        // Remove newline character from fgets

        int len = strlen(str);

        if (len > 0 && str[len - 1] == '\n')

        {

            str[len - 1] = '\0';

        }

    }

}

void displayAll(Players \*p, int n)

{

    for (int i = 0; i < n; i++)

    {

        printf("Name -> %s | ", p[i].name);

        printf("Matchs Played -> %d | ", p[i].noOfMatches);

        printf("Runs -> %d | ", p[i].runs);

        printf("Wickets -> %d\n", p[i].wickets);

    }

}

void displayByIndex(Players \*p, int i)

{

    printf("Name -> %s | ", p[i].name);

    printf("Matchs Played -> %d | ", p[i].noOfMatches);

    printf("Runs -> %d | ", p[i].runs);

    printf("Wickets -> %d\n", p[i].wickets);

}

void storeStruct(Players \*p, int n)

{

    for (int i = 0; i < n; i++)

    {

        printf("---Players %d---\n", i + 1);

        printf("Enter name of the Players\n");

        fgetsInput(p[i].name, sizeof(p[i].name));

        printf("Enter No of matching played\n");

        scanf("%d", &p[i].noOfMatches);

        printf("Enter runs score by player\n");

        scanf("%d", &p[i].runs);

        printf("Enter wickets taken by player\n");

        scanf("%d", &p[i].wickets);

    }

}

void displayTableToppers(Players \*p, int n)

{

    int maxRuns = p[0].runs, maxWicket = p[0].wickets;

    int maxRunsIndex = 0, maxWicketIndex = 0;

    for (int i = 0; i < n; i++)

    {

        if (p[i].runs > maxRuns)

        {

            maxRuns = p[i].runs;

            maxRunsIndex = i;

        }

        if (p[i].wickets > maxWicket)

        {

            maxRuns = p[i].wickets;

            maxWicketIndex = i;

        }

    }

    printf("Player who score maximum runs\n");

    displayByIndex(p, maxRunsIndex);

    printf("Player who taken maximum wickets\n");

    displayByIndex(p, maxWicketIndex);

}

int main()

{

    Players p[3];

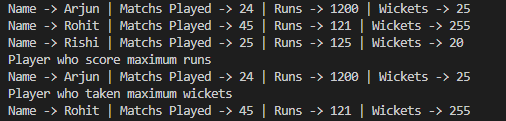
    storeStruct(p, 3);

    displayAll(p, 3);

    displayTableToppers(p, 3);

    return 0;

}



Q4) Point of Sale System: Build a simple point of sale system using structures to represent products with attributes like name, price, and quantity. Allow users to add items to a cart and calculate the total cost.

#include <stdio.h>

#include <string.h>

/\*

Point of Sale System: Build a simple point of sale system using structures to represent products with attributes like name, price, and quantity. Allow users to add items to a cart and calculate the total cost.

\*/

typedef struct Products

{

    int id;

    char name[30];

    double price;

} Products;

typedef struct CartItems

{

    Products p1;

    int quantity;

} CartItems;

void fgetsInput(char \*str, size\_t size)

{

    fflush(stdin);

    if (fgets(str, size, stdin))

    {

        // Remove newline character from fgets

        int len = strlen(str);

        if (len > 0 && str[len - 1] == '\n')

        {

            str[len - 1] = '\0';

        }

    }

}

void displayAll(Products \*p, int n)

{

    printf("----------------------------------\n");

    printf("|  Id  |    Product     | Price  |\n");

    printf("|------|----------------|--------|\n");

    for (int i = 0; i < n; i++)

    {

        int len = strlen(p[i].name);

        printf("|  %d   ", p[i].id);

        printf("| %s", p[i].name);

        for (int i = 1; i <= 15 - len; i++)

        {

            printf(" ");

        }

        i == 3 && printf("| %.2lf |\n", p[i].price);

        i != 3 && printf("| %.2lf  |\n", p[i].price);

    }

    printf("----------------------------------\n");

}

// void displayByIndex(Products \*p, int i)

// {

//     printf("Name -> %s | ", p[i].name);

//     printf("Matchs Played -> %d | ", p[i].noOfMatches);

//     printf("Runs -> %d | ", p[i].runs);

//     printf("Wickets -> %d\n", p[i].wickets);

// }

// void storeStruct(Players \*p, int n)

// {

//     for (int i = 0; i < n; i++)

//     {

//         printf("---Players %d---\n", i + 1);

//         printf("Enter name of the Players\n");

//         fgetsInput(p[i].name, sizeof(p[i].name));

//         printf("Enter No of matching played\n");

//         scanf("%d", &p[i].noOfMatches);

//         printf("Enter runs score by player\n");

//         scanf("%d", &p[i].runs);

//         printf("Enter wickets taken by player\n");

//         scanf("%d", &p[i].wickets);

//     }

// }

void generateAndDisplayBill(CartItems \*c, int\* n)

{

    double totalAmount = 0;

    printf("-------------------------------------------------------\n");

    printf("|  Id  |    Product     | Price  | Quantity |  Total  |\n");

    printf("|------|----------------|--------|----------|---------|\n");

    for (int i = 0; i < \*n; i++)

    {

        int len = strlen(c[i].p1.name);

        printf("|  %d   ", i+1);

        printf("| %s", c[i].p1.name);

        for (int i = 1; i <= 15 - len; i++)

        {

            printf(" ");

        }

        // printf("| %.2lf  \n", c[i].p1.price);

        c[i].p1.id == 3 && printf("| %.2lf ", c[i].p1.price);

        c[i].p1.id != 3 && printf("| %.2lf  ", c[i].p1.price);

        printf("|  %d      ", c[i].quantity);

        printf("| %.2lf |\n", c[i].quantity \* c[i].p1.price);

        totalAmount += c[i].quantity \* c[i].p1.price;

    }

    printf("-------------------------------------------------------\n");

    printf("                     Total Amount to pay ---> %.2lf\n\n\n", totalAmount);

}

void addToCart(CartItems \*c, Products \*p, int \*i)

{

    int choice=1, pid;

    while (choice)

    {

        printf("Enter Product id of product want to add to cart\n");

        scanf("%d", &pid);

        //adding product at index sent from main

        c[\*i].p1 = p[pid - 1];

        printf("Enter quantity\n");

        printf("%d %d  \n", \*i , c[\*i].quantity);

        (\*i)++;

        printf("\nPress 1 to Add more items\n");

        printf("Press 0 to generate bill\n");

        scanf("%d", &choice);

        if(choice) displayAll(p,5);

    }

    generateAndDisplayBill(c, &(\*i));

}

void hardCodedProducts(Products \*p)

{

    p[0].id = 1;

    strcpy(p[0].name, "Milk");

    p[0].price = 30;

    p[1].id = 2;

    strcpy(p[1].name, "Pen");

    p[1].price = 10;

    p[2].id = 3;

    strcpy(p[2].name, "Chips");

    p[2].price = 20;

    p[3].id = 4;

    strcpy(p[3].name, "Washing Powder");

    p[3].price = 150;

    p[4].id = 5;

    strcpy(p[4].name, "Soap");

    p[4].price = 20;

}

int main()

{

    int cartItemIndex = 0;

    Products p[5];

    CartItems c[50];

    // storeStruct(p, 3);

    hardCodedProducts(p);

    displayAll(p, 5);

    addToCart(c, p, &cartItemIndex);

    // displayTableToppers(p, 3);

    return 0;

}

