Assignment - 21 A Job Ready Bootcamp in C++, DSA and IOT MySirG

Structure

1. Define a structure Employee with member variables id, name, salary

#include <stdio.h>

struct Employee

{

int id;

float salary;

char name[20];

};

void displayUserData(struct Employee user1)

{

printf("\nUser id - %d", user1.id);

printf("\nUser name: %s", user1.name);

printf("User Salary: %.2f", user1.salary);

}

void inputUserData(struct Employee \*user1)

{

printf("Enter Id: ");

scanf("%d", &user1->id);

fflush(stdin); // because buffer is full so next fgets function not take name to we use fflush function

printf("Enter User Name: ");

fgets(user1->name, sizeof(user1->name), stdin);

printf("Enter User salary: ");

scanf("%f", &user1->salary);

}

int main()

{

struct Employee user1;

inputUserData(&user1);

displayUserData(user1);

return 0;

}

2. Write a function to take input employee data from the user. [ Refer structure from

question 1 ]

#include <stdio.h>

struct Employee

{

int id;

float salary;

char name[20];

};

void inputUserData(struct Employee user1)

{

printf("Enter Id: ");

scanf("%d", &user1.id);

fflush(stdin); // because buffer is full so next fgets function not take name to we use fflush function

printf("Enter User Name: ");

fgets(user1.name, sizeof(user1.name), stdin);

printf("Enter User salary: ");

scanf("%f", &user1.salary);

}

int main()

{

struct Employee user1;

inputUserData(user1);

return 0;

}

3. Write a function to display employee data. [ Refer structure from question 1 ]

#include <stdio.h>

struct Employee

{

int id;

char name[20];

float salary;

};

void displayUserData(struct Employee user1)

{

printf("\nUser1 Id: %d", user1.id);

printf("\nUser Name: %s", user1.name);

printf("User salary: %.2f", user1.salary);

}

void inputUserData(struct Employee \*user1)

{

printf("Enter Id: ");

scanf("%d", &user1->id);

fflush(stdin); // because buffer is full so next fgets function not take name to we use fflush function

printf("Enter User Name: ");

fgets(user1->name, sizeof(user1->name), stdin);

printf("Enter User salary: ");

scanf("%f", &user1->salary);

}

int main()

{

struct Employee user1;

inputUserData(&user1);

displayUserData(user1);

return 0;

}

4. Write a function to find the highest salary employee from a given array of 10

employees. [ Refer structure from question 1]

#include <stdio.h>

#include <string.h>

struct Employee

{

int id;

char name[20];

float salary;

};

void inputUserData(struct Employee users[], int arrLength)

{

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

{

printf("\nEnter %d User Details:\n",i+1);

printf("Enter User Id: ");

scanf("%d", &users[i].id);

fflush(stdin); // because buffer is full so next fgets function not take name to we use fflush function

printf("Enter User Name: ");

fgets(users[i].name, sizeof(users[i].name), stdin);

printf("Enter User salary: ");

scanf("%f", &users[i].salary);

}

}

void heighestSalaryEmp(struct Employee users[], int arrLength)

{

int mxSalaryEmpId = -1;

int ind = -1;

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

{

if (mxSalaryEmpId < users[i].salary)

{

mxSalaryEmpId = users[i].salary;

ind = i;

}

}

printf("\nHeighest Salary Employee id - %d and Employee Name - %s", users[ind].id, users[ind].name);

}

int main()

{

struct Employee users[10];

inputUserData(users, 10);

heighestSalaryEmp(users, 10);

return 0;

}

5. Write a function to sort employees according to their salaries [ refer structure from

question 1]

6. Write a function to sort employees according to their names [refer structure from

question 1]

7. Write a program to calculate the difference between two time periods.

#include <stdio.h>

struct timePeriod

{

int hrs, min, sec;

};

void inputPeriod(struct timePeriod \*time)

{

printf("\nEnter hourse: ");

scanf("%d", &time->hrs);

printf("\nEnter minute: ");

scanf("%d", &time->min);

printf("\nEnter second: ");

scanf("%d", &time->sec);

}

void diffTimePeriod(struct timePeriod time1, struct timePeriod time2)

{

int totalSec1 = time1.hrs \* 60 \* 60 + time1.min \* 60 + time1.sec;

int totalSec2 = time2.hrs \* 60 \* 60 + time2.min \* 60 + time2.sec;

int time = totalSec1 - totalSec2;

int hrs, min, sec;

sec = time % 60;

time /= 60;

min = time % 60;

time /= 60;

hrs = time;

printf("\nRemaming time\nhrs - %d, min - %d, sec - %d", hrs, min, sec);

}

int main()

{

struct timePeriod time1, time2;

inputPeriod(&time1);

inputPeriod(&time2);

diffTimePeriod(time1, time2);

return 0;

}

8. Write a program to store information of 10 students and display them using structure.

#include <stdio.h>

#include <string.h>

struct Student

{

int rollNo;

long int contact\_info;

char name[20];

};

void inputStudentData(struct Student stu[], int arrLength)

{

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

{

printf("\nEnter %d student Details:\n", i + 1);

printf("Enter student rollNo: ");

scanf("%d", &stu[i].rollNo);

fflush(stdin); // because buffer is full so next fgets function not take name to we use fflush function

printf("Enter student Name: ");

fgets(stu[i].name, sizeof(stu[i].name), stdin);

printf("Enter student contact number: ");

scanf("%d", &stu[i].contact\_info);

}

}

void displayStudentData(struct Student stu[], int arrLength)

{

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

{

printf("\n%d students details\n", i + 1);

printf("\nStudent Roll No: %d", stu[i].rollNo);

printf("\nStudents name: %s", stu[i].name);

printf("Students contact number: %d", stu[i].contact\_info);

}

}

int main()

{

struct Student stu[1];

inputStudentData(stu, 1);

displayStudentData(stu, 1);

return 0;

}

9. Write a program to store information of n students and display them using structure

#include <stdio.h>

#include <string.h>

struct Student

{

int rollNo;

long int contact\_info;

char name[20];

};

void inputStudentData(struct Student stu[], int arrLength)

{

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

{

printf("\nEnter %d student Details:\n", i + 1);

printf("Enter student rollNo: ");

scanf("%d", &stu[i].rollNo);

fflush(stdin); // because buffer is full so next fgets function not take name to we use fflush function

printf("Enter student Name: ");

fgets(stu[i].name, sizeof(stu[i].name), stdin);

printf("Enter student contact number: ");

scanf("%d", &stu[i].contact\_info);

}

}

void displayStudentData(struct Student stu[], int arrLength)

{

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

{

printf("\n\n%d students details", i + 1);

printf("\nStudent Roll No: %d", stu[i].rollNo);

printf("\nStudents name: %s", stu[i].name);

printf("Students contact number: %d", stu[i].contact\_info);

}

}

int main()

{

int totalStu;

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

scanf("%d", &totalStu);

struct Student stu[totalStu];

inputStudentData(stu, totalStu);

displayStudentData(stu, totalStu);

return 0;

}

10. Write a program to enter the marks of 5 students in Chemistry, Mathematics and

Physics (each out of 100) using a structure named Marks having elements roll no.,

name, chem\_marks, maths\_marks and phy\_marks and then display the percentage

of each student.

#include <stdio.h>

struct studentMarks

{

int rollNo;

char name[20];

int chem\_marks, phy\_marks, maths\_marks;

};

void inputStuMark(struct studentMarks student[], int arrLength)

{

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

{

printf("\n\nEnter %d Student details: ", i+1);

printf("\nEnter rollNo: ");

scanf("%d", &student[i].rollNo);

fflush(stdin);

printf("Enter Name of student: ");

fgets(student[i].name, 20, stdin);

printf("Enter chemistry, physics and maths marks(out of 100): ");

scanf("%d%d%d", &student[i].chem\_marks, &student[i].phy\_marks, &student[i].maths\_marks);

}

}

void studentPercentage(struct studentMarks student[], int arrlength)

{

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

{

printf("\n\n%d student Details: ", i+1);

printf("\nRollNo - %d", student[i].rollNo);

printf("\nName: %s", student[i].name);

printf("chem\_marks - %d, phy\_marks - %d, math\_marks - %d", student[i].chem\_marks, student[i].phy\_marks,student[i].maths\_marks);

float average = (student[i].chem\_marks + student[i].maths\_marks + student[i].phy\_marks)/3.0f;

printf("\nAverage of marks: %.2f",average);

}

}

int main()

{

struct studentMarks student[5];

inputStuMark(student, 5);

studentPercentage(student, 5);

return 0;

}