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 Empoyee

{

int id;

char name[20];

float salary;

};

int main()

{

struct Employee;

printf("\n");

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;

char name[20];

float salary;

};

struct Employee input()

{

struct Employee m;

printf("Enter the Employee Id, Name & Salary\n");

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

getchar(); // clear the newline character from input buffer

fgets(m.name,20,stdin);

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

return m;

}

int main()

{

struct Employee m;

m = input();

printf("Employee Id: %d\n", m.id);

printf("Employee Name: %s", m.name);

printf("Employee Salary: %f\n", m.salary);

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;

};

struct Employee input()

{

struct Employee m;

printf("Enter the Employee Id, Name & Salary\n");

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

getchar(); // clear the newline character from input buffer

fgets(m.name,20,stdin);

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

return m;

}

void display(struct Employee m)

{

printf("Employee Id: %d\n", m.id);

printf("Employee Name: %s", m.name);

printf("Employee Salary: %f\n", m.salary);

}

int main()

{

struct Employee m;

m = input();

display(m);

printf("\n");

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<stdlib.h>

//global variable

int i;

struct Employee

{

int id;

char name[20];

float salary;

};

struct Employee input(struct Employee \*x)

{

printf("Enter no.%d Employee Id, Name & Salary\n",i+1);

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

getchar(); // clear the newline character from input buffer

fgets(x->name, 20, stdin);

fflush(stdin);

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

return \*x;

}

void display(struct Employee m)

{

printf("no.%d\n",i+1);

printf("Employee Id: %d\n", m.id);

printf("Employee Name: %s", m.name);

printf("Employee Salary: %f\n", m.salary);

}

int compare(const void \*a, const void \*b)

{

const struct Employee \*ass = (const struct Employee \*)a;

const struct Employee \*bss = (const struct Employee \*)b;

if(ass->salary==bss->salary)

return 0;

else if(ass->salary<bss->salary)

return (-1);

else

return 1;

}

int main()

{

int n;

printf("Enter the size of the Employee list:\n");

scanf("%d",&n);

struct Employee m[n];

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

input(&m[i]);

printf("Before sorting:\n");

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

display(m[i]);

qsort(m,n,sizeof(struct Employee),compare);

printf("After sorting:\n");

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

display(m[i]);

printf("\n");

return 0;

}

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

question 1]

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

struct Employee

{

int id;

char name[20];

float salary;

};

void input(struct Employee \*emp)

{

printf("Enter Employee Id, Name & Salary:\n");

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

getchar();

fgets(emp->name, 20, stdin);

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

}

void display(struct Employee emp, int i)

{

printf("no.%d\n", i + 1);

printf("Employee Id: %d\n", emp.id);

printf("Employee Name: %s", emp.name);

printf("Employee Salary: %f\n", emp.salary);

}

void sort\_salary(struct Employee m[], int n)

{

for (int i = 0; i < n - 1; i++)

{

for (int j = i + 1; j < n; j++)

{

if (m[j].salary < m[i].salary)

{

struct Employee temp = m[i];

m[i] = m[j];

m[j] = temp;

}

}

}

}

int main()

{

struct Employee m[3];

printf("Enter details of 10 Employee:\n");

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

{

input(&m[i]);

}

// Displaying the array elements before sorting them using bubble sort algorithm

printf("Before sorting\n");

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

{

display(m[i], i);

printf("\n");

}

sort\_salary(m, 3);

printf("After sorting\n");

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

{

display(m[i], i);

printf("\n");

}

return 0;

}

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

question 1]

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

struct Employee

{

int id;

char name[20];

float salary;

};

void input(struct Employee \*emp)

{

printf("Enter Employee Id, Name & Salary:\n");

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

getchar();

fgets(emp->name, 20, stdin);

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

}

void display(struct Employee emp)

{

printf("Employee Id: %d\n", emp.id);

printf("Employee Name: %s\n", emp.name);

printf("Employee Salary: %f\n", emp.salary);

}

void sort\_name(struct Employee m[], int n)

{

for (int i = 0; i < n - 1; i++)

{

for (int j = i + 1; j < n; j++)

{

if (strcmp(m[j].name, m[i].name) < 0)

{

struct Employee temp = m[i];

m[i] = m[j];

m[j] = temp;

}

}

}

}

int main()

{

printf("Enter number of emplyoee:\n");

int n;

scanf("%d", &n);

// Clear input buffer;

int c;

while ((c = getchar()) != '\n' && c != EOF);

struct Employee m[n];

printf("Enter details of %d employee:\n", n);

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

input(&m[i]);

printf("Before sorting:\n");

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

{

printf("no. ;%d\n",i+1);

display(m[i]);

}

sort\_name(m, n);

printf("After sorting:\n");

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

{

printf("no. %d\n",i+1);

display(m[i]);

}

return 0;

}

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

#include <stdio.h>

struct Time

{

int hours;

int minutes;

int seconds;

};

struct Time getTimeDifference(struct Time start, struct Time end)

{

struct Time diff;

if (end.seconds < start.seconds)

{

end.seconds += 60;

end.minutes -= 1;

}

diff.seconds = end.seconds - start.seconds;

if (end.minutes < start.minutes)

{

end.minutes += 60;

end.hours -= 1;

}

diff.minutes = end.minutes - start.minutes;

diff.hours = end.hours - start.hours;

return diff;

}

int main()

{

struct Time start, end;

printf("Enter start time (hours minutes seconds): ");

scanf("%d %d %d", &start.hours, &start.minutes, &start.seconds);

printf("Enter end time (hours minutes seconds): ");

scanf("%d %d %d", &end.hours, &end.minutes, &end.seconds);

struct Time difference = getTimeDifference(start, end);

printf("Time Difference: %d hours, %d minutes, %d seconds\n",

difference.hours, difference.minutes, difference.seconds);

return 0;

}

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

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

struct student

{

int roll\_no;

char name[20];

float chem\_marks,math\_marks,phy\_marks;

};

void input(struct student \*st)

{

printf("Enter Student Roll:\n");

scanf("%d",&st->roll\_no);

printf("Enter Studnet Name:\n");

fflush(stdin);

fgets(st->name, 20, stdin);

st->name[sizeof(st->name)-1]='\0';

printf("Enter Marks in Chemistry:\n");

scanf("%f",&st->chem\_marks);

printf("Enter marks in Maths:\n");

scanf("%f",&st->math\_marks);

printf("Enter marks in Physics:\n");

scanf("%f",&st->phy\_marks);

}

void display(struct student st, int i)

{

printf("no.%d\n", i + 1);

printf("Student Roll Number: %d\n", st.roll\_no);

printf("Student Name: %s", st.name);

printf("Marks in Chemistry: %f\n", st.chem\_marks);

printf("Marks in Maths: %f\n",st.math\_marks);

printf("Marks in Physics: %f\n",st.phy\_marks);

}

int main()

{

struct student list[10];

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

input(&list[i]);

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

display(list[i],i);

return 0;

}

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

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

struct student

{

int roll\_no;

char name[20];

float chem\_marks,math\_marks,phy\_marks;

};

void input(struct student \*st)

{

printf("Enter Student Roll:\n");

scanf("%d",&st->roll\_no);

printf("Enter Studnet Name:\n");

fflush(stdin);

fgets(st->name, 20, stdin);

st->name[sizeof(st->name)-1]='\0';

printf("Enter Marks in Chemistry:\n");

scanf("%f",&st->chem\_marks);

printf("Enter marks in Maths:\n");

scanf("%f",&st->math\_marks);

printf("Enter marks in Physics:\n");

scanf("%f",&st->phy\_marks);

}

void display(struct student st, int i)

{

printf("no.%d\n", i + 1);

printf("Student Roll Number: %d\n", st.roll\_no);

printf("Student Name: %s", st.name);

printf("Marks in Chemistry: %f\n", st.chem\_marks);

printf("Marks in Maths: %f\n",st.math\_marks);

printf("Marks in Physics: %f\n",st.phy\_marks);

}

int main()

{

int n;

printf("Enter the number of student:\n");

scanf("%d",&n);

struct student list[n];

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

input(&list[i]);

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

display(list[i],i);

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>

#include<string.h>

#include<stdlib.h>

struct student

{

int roll\_no;

char name[20];

float chem\_marks,math\_marks,phy\_marks;

};

void input(struct student \*st)

{

printf("Enter Student Roll:\n");

scanf("%d",&st->roll\_no);

printf("Enter Studnet Name:\n");

fflush(stdin);

fgets(st->name, 20, stdin);

st->name[sizeof(st->name)-1]='\0';

printf("Enter Marks in Chemistry:\n");

scanf("%f",&st->chem\_marks);

printf("Enter marks in Maths:\n");

scanf("%f",&st->math\_marks);

printf("Enter marks in Physics:\n");

scanf("%f",&st->phy\_marks);

}

void display(struct student st, int i)

{

printf("no.%d\n", i + 1);

printf("Student Roll Number: %d\n", st.roll\_no);

printf("Student Name: %s", st.name);

printf("Marks in Chemistry: %f\n", st.chem\_marks);

printf("Marks in Maths: %f\n",st.math\_marks);

printf("Marks in Physics: %f\n",st.phy\_marks);

}

int main()

{

struct student list[5];

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

input(&list[i]);

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

display(list[i],i);

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

{

float sum=(list[i].chem\_marks+list[i].math\_marks+list[i].phy\_marks);

printf("The average marks obtained by %s is %f\n",list[i].name,sum/3);

}

return 0;

}