LAB8: C Structures

Task1(1 structure – 2 record – array notation): Write a program stores the information (name and marks) of 5 students using structures.

In this program, a structure, student is created.

This structure has two members: name (string) and marks (float).

Then, we created a structure array of size 5 to store information of 5 students.

Using for loop, the program takes the information of 5 students from the user and displays it on the screen.

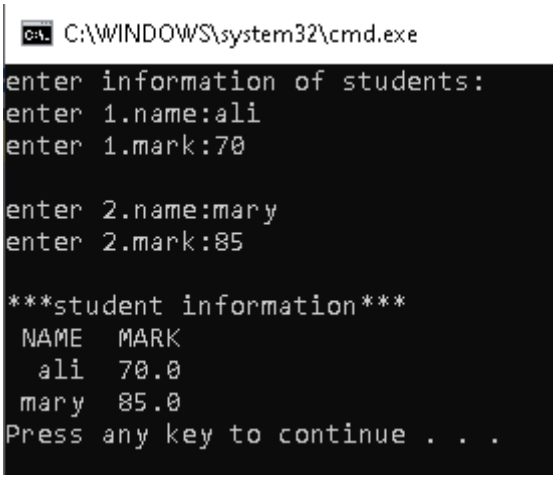
Sol:

```
#include "stdafx.h"
struct student
{
	char name[50];
	float marks;
} s[2];

void main()
{
	char b[2];
	int i;
	printf("enter information of students:\n");
	for ( i = 0; i < 2; i++)
	{
		printf("enter %d.name:",i+1);
		gets(s[i].name);

		printf("enter %d.mark:",i+1);
		scanf_s("%f", &s[i].marks);

		gets(b);
		printf("\n");
	}
	printf("***student information***\n");
	printf("%5s%6s\n", "NAME", "MARK");
	for (i = 0; i < 2; i++)
	{
		printf("%5s%6.1f\n",s[i].name,s[i].marks);
	}
}
```



Task2 (nested structure – 1 record): Design structures to store information on customers and their bank accounts based on the following form.

Customer
- **Id**
- **Name**

Account
- **Accno**
- **Balance**
- **Customer information**

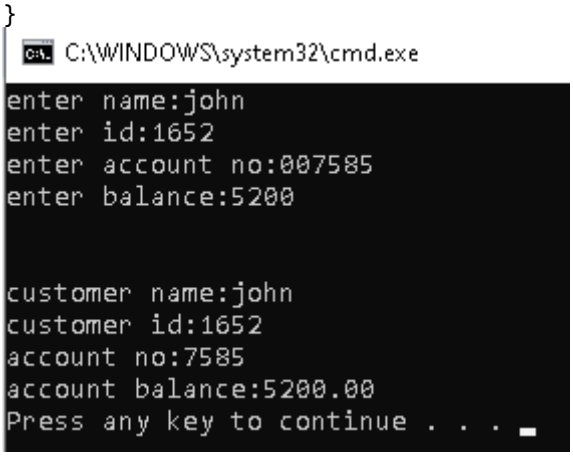
```
#include "stdafx.h"
struct customer
{
	int id;
	char name[20];
};

struct account
{
	int accno;
	float balance;
	struct customer cinfo;
};

void main()
{
	char b[2];
	struct account a;
	printf("enter name:");
	gets(a.cinfo.name);
	printf("enter id:");
	scanf_s("%d", &a.cinfo.id);
	gets(b);
	printf("enter account no:");
	scanf_s("%d", &a.accno);

	printf("enter balance:");
	scanf_s("%f", &a.balance);
```

```
	printf("\n\ncustomer name:%s\ncustomer id:%d\n",a.cinfo.name,a.cinfo.id);
	printf("account no:%d\naccount balance:%.2f\n", a.accno, a.balance);
}
```



Task3A (nested structure – 1 record – pointer notation): Modification of Task2

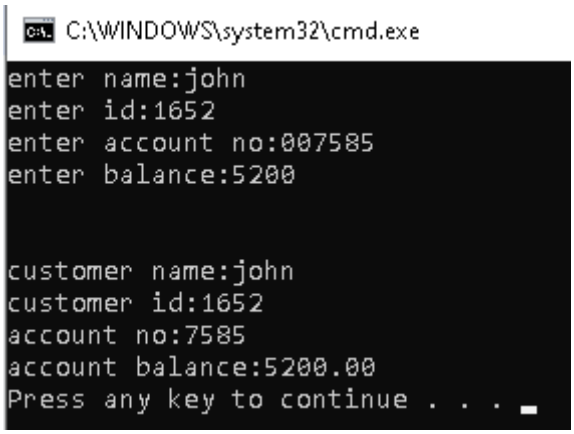
```
#include "stdafx.h"
struct customer
{
	int id;
	char name[20];
};

struct account
{
	int accno;
	float balance;
	struct customer cinfo;
};

void main()
{
	char b[2];
	struct account a,*p;
	p = &a;
	printf("enter name:");
	gets((*p).cinfo.name);
	printf("enter id:");
	scanf_s("%d", &(p)->cinfo.id);
	gets(b);
	printf("enter account no:");
	scanf_s("%d", &(*p).accno);

	printf("enter balance:");
	scanf_s("%f", &(p)->balance);

	printf("\n\ncustomer name:%s\ncustomer id:%d\n",(*p).cinfo.name,(p)->cinfo.id);
	printf("account no:%d\naccount balance:%.2f\n", (*p).accno, (p)->balance);
}
```



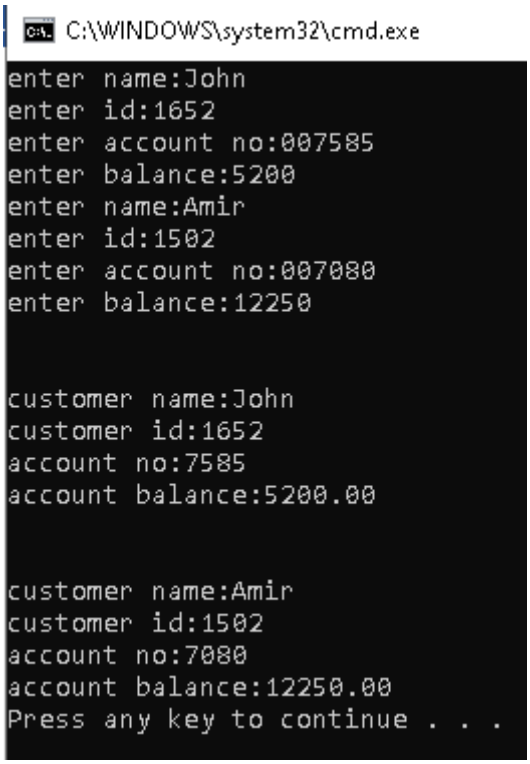
Task3B(nested structure – 2 record – pointer notation): Modification of Task3A

```
#include "stdafx.h"
struct customer
{
	int id;
	char name[20];
};

struct account
{
	int accno;
	float balance;
	struct customer cinfo;
};

void main()
{
	char b[2]; int i;
	struct account a[2],*p;
	p = &a;
	for (i = 0; i < 2; i++)
	{
		printf("enter name:");
		gets((*(p+i)).cinfo.name);
		printf("enter id:");
		scanf_s("%d", &(p+i)->cinfo.id);
		gets(b);
		printf("enter account no:");
		scanf_s("%d", &(*(p+i)).accno);

		printf("enter balance:");
		scanf_s("%f", &(p+i)->balance);
		gets(b);
	}
	for (i = 0; i < 2; i++)
	{
		printf("\n\ncustomer name:%s\ncustomer id:%d\n", (*(p+i)).cinfo.name,
(p+i)->cinfo.id);
		printf("account no:%d\naccount balance:%.2f\n", (*(p+i)).accno, (p+i)-
>balance);
	}
}
```



Task4(nested structure – 2 record – pointer notation-printing information in a function):
Modification of Task3

```
#include "stdafx.h"
struct customer
{
	int id;
	char name[20];
};

struct account
{
	int accno;
	float balance;
	struct customer cinfo;
};
void printfInfo(struct account *p)
{
	for (int i = 0; i < 2; i++)
	{
		printf("\n\ncustomer name:%s\ncustomer id:%d\n", (*(p + i)).cinfo.name,
(p + i)->cinfo.id);
		printf("account no:%d\naccount balance:%.2f\n", (*(p + i)).accno, (p +
i)->balance);
	}
}
void main()
{
	char b[2]; int i;
	struct account a[2],*p;

	p = &a;
	for (i = 0; i < 2; i++)
```

```
	{
		printf("enter name:");
		gets((*(p+i)).cinfo.name);
		printf("enter id:");
		scanf_s("%d", &(p+i)->cinfo.id);
		gets(b);
		printf("enter account no:");
		scanf_s("%d", &(*(p+i)).accno);

		printf("enter balance:");
		scanf_s("%f", &(p+i)->balance);
		gets(b);
	}
	printfInfo(a);
}
```