**Programming Fundamentals**

**Lab 13**

**Submitted To:**

Mr. Dilshad Sabir

**Submitted By:**

Manaal Waseem

FA18-BCE-074

**In Lab:**

**Task 1:**

**Write a C program that initialize a structure called “mobile”. Instance of that object should contain fields vendor i.e. Huawei, Samsung or Nokia, owner name and its IMIE (International Mobile Equipment Identity) number. User should enter the minimum of two records. And print these records using pointers.**

**Program:** In this program, user is prompted to enter data i.e. **‘vendor’**, **‘owner’** and **‘imie’** for two mobiles. This data is stored in structure **‘struct mobile’** declared globally. This structure is passed as pointer to **‘print\_mobile’** function. The details for all two mobiles are printed as output on the console when user-defined function **‘print\_mobile’** is called in main.

1 #include <stdio.h>

2 #include <stdlib.h>

3

4 **struct** mobile{

5 **char** vendor[8];

6 **char** owner[20];

7 **char** imie[15];

8 };

9

10 **void** print\_mobile(**struct** mobile \* p, **int** size);

11 **int** main()

12 {

13 **struct** mobile phone[2];

14 **struct** mobile \*ptr\_phone = &phone[0];

15

16 **for**(**int** i=0; i<2; i++)

17 {

18 printf("\nPlease enter data for mobile#%d\n",i+1);

19 printf("vendor: ");

20 gets(phone[i].vendor);

21 printf("\n");

22 printf("Owner Name: ");

23 gets(phone[i].owner);

24 printf("\n");

25 printf("IMIE Number: ");

26 gets(phone[i].imie);

27 printf("\n");

28 }

29 print\_mobile(ptr\_phone,2);

30 **return** 0;

31 }

32

33 **void** print\_mobile(**struct** mobile \* p, **int** size)

34 {

35 **for**(**int** i=0; i<size; i++)

36 {

37 printf("\nMobile#%d\n\n",i+1);

38 printf("vendor: ");

39 puts((p+i)->vendor);

40 printf("\n");

41 printf("Owner Name: ");

42 puts((p+i)->owner);

43 printf("\n");

44 printf("IMIE Number: ");

45 puts((p+i)->imie);

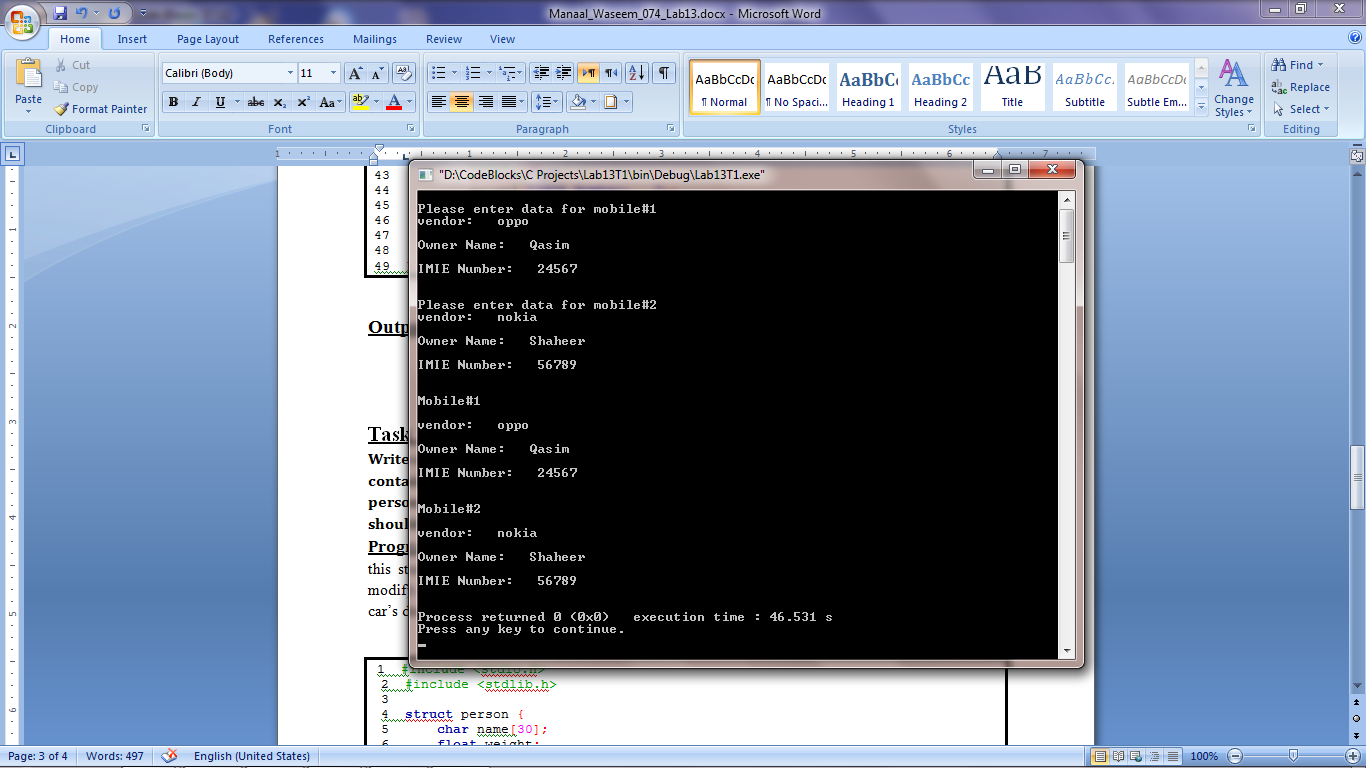
46 printf("\n");

47

48 }

49 }

**Output:**



**Task 2:**

**Write a C program that has structure called person. Instance of that structure should contain name, weight and age of that person. Program should take number of persons. Then use dynamic memory allocation to have that memory for record. User should enter the minimum specified number of records. And print these records.**

**Program:** In this program, a structure **‘struct person’** is declared globally to store data i.e. **‘name’**, **‘weight’** and **‘age’** for the number of persons specified by the user. A pointer **‘ptr\_person’** is declared. Now we dynamically allocate a block of memory to store data for the number of person specified by the user. Now user is prompted to enter data i.e. **‘name’**, **‘weight’** and **‘age’** for the number of persons specified by the user. The details are then printed as output on the console.

1 #include <stdio.h>

2 #include <stdlib.h>

3

4 **struct** person {

5 **char** name[30];

6 **float** weight;

7 **int** age;

8 };

9

10 **int** main()

11 {

12 **struct** person \*ptr\_person;

13 **int** i, n;

14

15 printf("Enter number of persons: ");

16 scanf("%d", &n);

17

18 ptr\_person = (**struct** person\*) malloc(n \* **sizeof**(**struct** person));

19

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

21 {

22 printf("Enter first name, weight and age respectively: ");

23 scanf("%s%f%d", &(ptr\_person+i)->name, &(ptr\_person+i)->weight,&(ptr\_person+i)->age);

24 }

25

26 printf("Printing Information:\n");

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

28 {

29 printf("\nPerson#%d:\n",i+1);

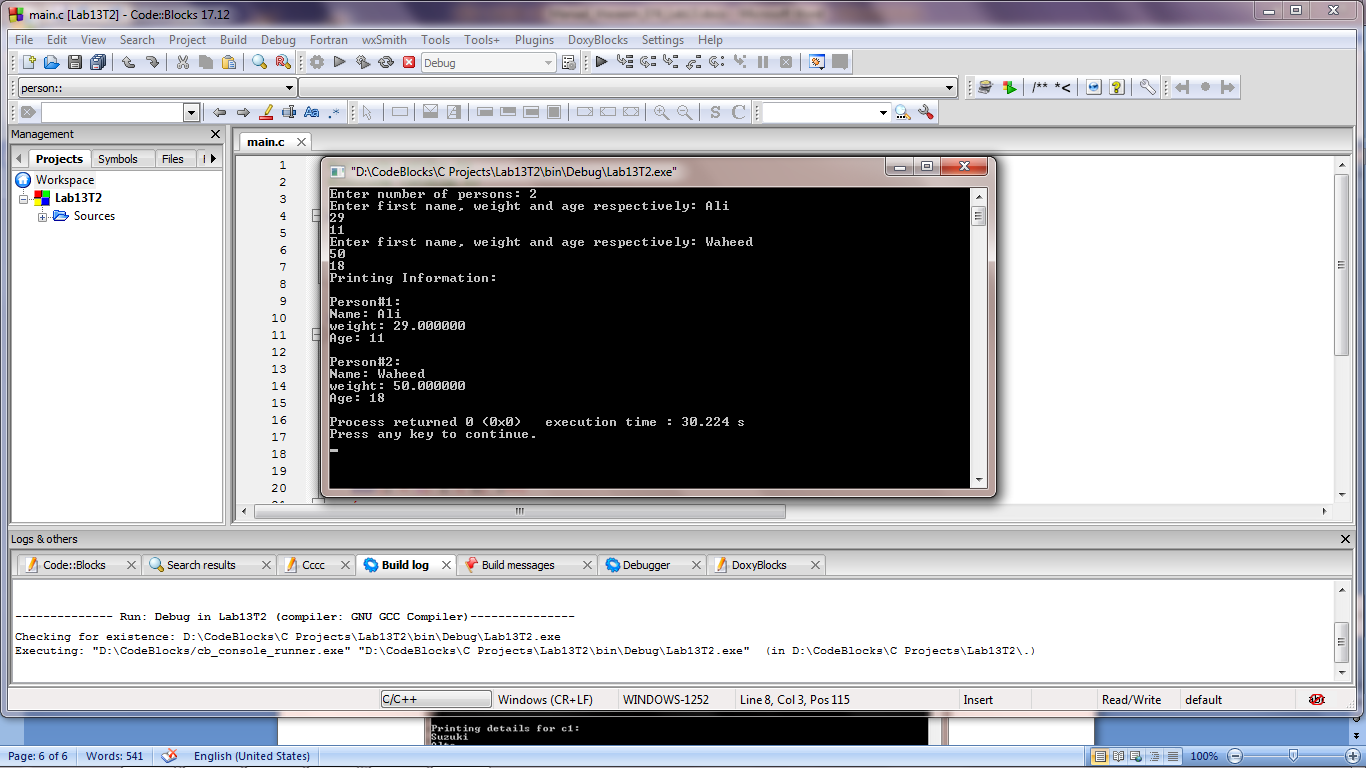
30 printf("Name: %s\nweight: %f\nAge: %d\n", (ptr\_person+i)->name, (ptr\_person+i)->weight, (ptr\_person+i)->age);

31 }

32 **return** 0;

33 }

**Output:**



**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**THE END**