Foundations of Technical Programming
Week 10

You may be asked to demonstrate/explain your work to the tutor, if you are absent/unavailable or fail to demonstrate properly, zero marks will be awarded.

Sample1:
The following sample program uses a structure “Accessories” as a nested structure with the structure “Toy” to read and display the values of three (3) toys. Each toy is to have one accessory.

```c
#include <stdio.h>
#include <stdlib.h>

typedef struct Accessories {
    int accID;
    char accName[10];
}Accessories;

typedef struct Toy {
    int ID;
    char name[10];
    Accessories extra;  // Structire toy has a variable of the type Accessories
    float price;

}Toy;

int main()
{
    Toy a[3];   //a is an array of type toy size 3
    int i;
    for(i=0; i<3; i++)
    {
        printf("Enter toy details... ID, Name & Price \n");
        scanf("%d", &a[i].ID);
        fflush(stdin);
        gets(a[i].name);
        fflush(stdin);
        scanf("%f", &a[i].price);
        fflush(stdin);
        printf("Enter %s Accessories details ... ID & Name \n");
        scanf("%d", &a[i].extra.accID);
        fflush(stdin);
        gets(a[i].extra.accName);
    }

    system("cls");
    printf("Toy Details \n");
    for(i=0; i<3; i++)
        printf("Toy ID - %d Name %s Price %.2f Acc-ID %d Acc-Name %s \n",
        a[i].ID, a[i].name, a[i]. price, a[i].extra.accID, a[i].extra.accName);

    return 0;
}
```

Sample Output

```
Toy Details
Toy ID - 101 Name Barbie Price 55.00 Acc-ID 1 Acc-Name Dress
Toy ID - 102 Name Spiderman Price 65.00 Acc-ID 1 Acc-Name Mask
Toy ID - 103 Name Dora Price 44.00 Acc-ID 1 Acc-Name Backbag
```

Task 10.1 – Nested Structure

Aim: To demonstrate your knowledge in using nested structures

Title: C Program to Store Information of Students Using Structure

Write C-Program based on the following instructions

a. Create a structure subject – the structure to contain **subjectID, subjectName** and **marks.**
b. Create a structure student – the structure to contain **studentID, studentName, subjects** (variable subjects should be of the type **structure subject). Each student to be enrolled in three (3) subjects.** **Hint – Create an array of subject of size 3.**
c. Read details for one (1) student… including three (3) subject details
d. Display the values of the student along with the details of three subjects
e. The output of your program should match the following sample output

```
Student details
ID - S1001        Name John

Subject ID      Name            Marks
COS10008        C-Prog          65.00
COS10020        Web             54.00
COS10011        IBIS            89.00
```

Sample2:
The following sample program uses a structure “Accessories” as a nested structure with the structure “Toy” to read and display the values of three (3) toys. Each toy is to have one accessory. The program uses a function named “printDetails(Toy[], int size)” to print the details of the Toys.

```c
#include <stdio.h>
#include <stdlib.h>

typedef struct Accessories {
    int accID;
    char accName[10];
}Accessories;

typedef struct Toy {
    int ID;
    char name[10];
    Accessories extra;  // Structire toy has a variable of the type Accessories
    float price;

}Toy;

void printDetails(Toy a[], int size)
{
    int i;
    system("cls");
    printf("Toy Details \n");
    for(i=0; i<3; i++)
        printf("Toy ID - %d Name %s Price %.2f Acc-ID %d Acc-Name %s \n",
        a[i].ID, a[i].name, a[i]. price, a[i].extra.accID, a[i].extra.accName);
}
int main()
{
    Toy a[3];   //a is an array of type toy size 3
    int i;
    for(i=0; i<3; i++)
    {
        printf("Enter toy details... ID, Name & Price \n");
        scanf("%d", &a[i].ID);
        fflush(stdin);
        gets(a[i].name);
        fflush(stdin);
        scanf("%f", &a[i].price);
        fflush(stdin);
        printf("Enter %s Accessories details ... ID & Name \n");
        scanf("%d", &a[i].extra.accID);
        fflush(stdin);
        gets(a[i].extra.accName);
    }
    //function printDetails called passing the array "a" and the size of the array (3)
    printDetails(a, 3);

    return 0;
}
```

Task 10.2 – Nested Structure

Aim: To demonstrate your knowledge in using nested structures and functions

Title: C Program to Store Information of Students Using Structure

Convert your Task 10.1 program based on the following instructions

a. Create a function displayDetails(student []) and print the details of the students.

Sample3:
The following sample program uses an enumeration “stock” to maintain the status of the stock (IN-Available & OUT-not available).
The enumeration is used inside the structure “Toy”.

```c
#include <stdio.h>
#include <stdlib.h>

typedef enum {IN, OUT}stock;
typedef struct Accessories {
    int accID;
    char accName[10];
}Accessories;

typedef struct Toy {
    int ID;
    char name[10];
    Accessories extra;  // Structire toy has a variable of the type Accessories
    float price;
    stock status;   //status can be IN or OUT
}Toy;

void printDetails(Toy a[], int size)
{
    int i;
    system("cls");
    printf("Toy Details \n");
    for(i=0; i<3; i++){
        printf("Toy ID - %d Name %s Price %.2f Acc-ID %d Acc-Name %s \n",
        a[i].ID, a[i].name, a[i]. price, a[i].extra.accID, a[i].extra.accName);
        switch(a[i].status)
        {
            case 0: printf("Toy available in store \n");break;
            case 1: printf("Toy unavailable in store \n"); break;
        }
    }
}
int main()
{
    Toy a[3];   //a is an array of type toy size 3
    int i;
    for(i=0; i<3; i++)
    {
        printf("Enter toy details... ID, Name & Price \n");
        scanf("%d", &a[i].ID);
        fflush(stdin);
        gets(a[i].name);
        fflush(stdin);
        scanf("%f", &a[i].price);
        fflush(stdin);
        printf("Enter %s Accessories details ... ID & Name \n");
        scanf("%d", &a[i].extra.accID);
        fflush(stdin);
        gets(a[i].extra.accName);
        printf("Enter 0 for stock available \nEnter 1 for out of stock \n");
        scanf("%d", &a[i].status);

    }
    //function printDetails called passing the array "a" and the size of the array (3)
    printDetails(a, 3);

    return 0;
}
```

Sample-Input Screen

```
Enter toy details... ID, Name & Price
101
Barbie
44
Enter  Accessories details ... ID & Name
1
Dress
Enter 0 for stock available
Enter 1 for out of stock
0
Enter toy details... ID, Name & Price
```

Sample-Output Screen

```
Toy Details
Toy ID - 101 Name Barbie Price 44.00 Acc-ID 1 Acc-Name Dress
Toy available in store
Toy ID - 102 Name Spiderman Price 54.00 Acc-ID 1 Acc-Name Mask
Toy unavailable in store
Toy ID - 103 Name Dora Price 76.00 Acc-ID 1 Acc-Name Backbag
Toy available in store
```

Task 10.3 – Enumeration

Aim: To demonstrate your knowledge in using enumeration

Title: C Program to Store Information of Students Using Structure

Convert your Task 10.2 program based on the following instructions

a. Using enumeration, create a user-defined type “Result” with two possible values “PASS” and “FAIL”.
b. Use the enumeration “Result” to create a variable “Status” in the structure student.
c. Read values for one student with three (3) subjects.
d. Assign the value “PASS” to the status – if the student has scored 50 or above in all three subjects. Otherwise, assign the value “FAIL” to the status.
e. Display the student details as shown below

Sample Input

```
Enter the student details ID and Name...
S101
John
Enter 1 Subject details ID - Name & Marks...
COS10008
C-Prog
54
Enter 2 Subject details ID - Name & Marks...
COS10020
Web
44
Enter 3 Subject details ID - Name & Marks...
COS10011
IBIS
89
```

Sample Output

```
Student details
ID - S101        Name John

Subject ID      Name            Marks
COS10008        C-Prog          54.00
COS10020        Web             44.00
COS10011        IBIS            89.00
Student is unsuccessful this study period
```