CSE115L – Programming Language I Lab

Lab - 25
Structure (User Defined Types)

In this lab, we will solve a few problems using user defined data types, namely structures. The following examples will help you remember the syntax.

Example 1: Declaring structure variables separately

```
#include <stdio.h>
#include<string.h>
struct book_data
{
    char title[100];
    char author[100];
    char topic[100];
    int id;
};

int main()
{
    struct book_data b;
    strcpy(b.title, "Title");
    strcpy(b.author, "Author");
    strcpy(b.topic, "Topic");
    b.id = 12;

    return 0;
}
```

Example 2: Declaring structure variables with structure definition

```
#include<stdio.h>
#include<string.h>
struct book_data
{
    char title[100];
    char author[100];
    char topic[100];
    int id;
}b;


int main()
{
    strcpy(b.author, "Author");
    strcpy(b.title, "Title");
    strcpy(b.topic, "Topic");
    b.id = 12;

    return 0;
}
```

Example 3: Array of Structures

```
#include <stdio.h>

struct student{
    char name[50];
    int roll;
    float marks;
}s[10];

void main(){
    int i;
    printf("Enter information of students:\n");

    for(i=0;i<10;++i)
    {
        s[i].roll=i+1;
        printf("\nFor roll number %d\n",s[i].roll);
        printf("Enter name: ");
        scanf("%s",s[i].name);
        printf("Enter marks: ");
        scanf("%f",&s[i].marks);
        printf("\n");
    }

    printf("Displaying information of students:\n\n");

    for(i=0;i<10;++i)
    {
        printf("\nInformation for roll number %d:\n",i+1);
        printf("Name: ");
        puts(s[i].name);
        printf("Marks: %.1f",s[i].marks);
   }
}
```

Example 4: C Program that reads two distances (in feet+inches) and prints their sum:

```
#include <stdio.h>
struct Distance{
    int feet;
    float inch;
}d1,d2,sum;

int main(){
    printf("1st distance\n");
    printf("Enter feet: ");
    scanf("%d",&d1.feet);  /* input of feet for structure variable d1 */
    printf("Enter inch: ");
    scanf("%f",&d1.inch);  /* input of inch for structure variable d1 */
    printf("2nd distance\n");
    printf("Enter feet: ");
    scanf("%d",&d2.feet);  /* input of feet for structure variable d2 */
    printf("Enter inch: ");
    scanf("%f",&d2.inch);  /* input of inch for structure variable d2 */

    sum.feet=d1.feet+d2.feet;
    sum.inch=d1.inch+d2.inch;

    if (sum.inch>12){  //If inch is greater than 12, changing it to feet.
        ++sum.feet;
        sum.inch=sum.inch-12;
    }
    printf("Sum of distances=%d\'-%.1f\"",sum.feet,sum.inch);
}
```

Example 5: Passing an array of Structures as function arguments (book records):

```
#include <stdio.h>
#include <string.h>
#define MAX_BOOKS 1000

int NUM_BOOKS=0; //global variable containing the actual number of books

struct Books
{
    char  title[50];
    char  author[50];
    char  subject[100];
    int   book_id;
} ;

void readBooks( struct Books b[] )
{
    /* read book specifications from user user until s/he enters empty
string as title*/
    int i;
    for(i=0; i < MAX_BOOKS; i++)    {
        printf("Enter book title (press just enter to finish): ");
        gets(b[i].title);
        if(strcmp(b[i].title, "")==0) break;
        printf("Enter author-names: ");
        gets(b[i].author);
        printf("Enter subject: ");
        gets(b[i].subject);
        printf("Enter id: ");
        scanf("%d", &b[i].book_id);
        fflush(stdin);
        NUM_BOOKS++; //update the number of books we have
    }
}

void printBooks( struct Books b[] )
{
    int i;
    printf("\n\n We have the following books:\n\n");
    for(i=0; i < NUM_BOOKS; i++)    {
        printf( "Book title : %s\n", b[i].title);
        printf( "Book author : %s\n", b[i].author);
        printf( "Book subject : %s\n", b[i].subject);
        printf( "Book book_id : %d\n\n", b[i].book_id);
    }
}
```

Perform the following tasks.

Task 1: The Manhattan distance between two points $P(x_1,y_1)$ and $Q(x_2,y_2)$ is defined as follows:

$$M.D. = |x_1-x_2| + |y_1-y_2|$$

- Create a structure that models a point in the 2-dimensional space.
- Using the above structure, take input for two points and calculate the Manhattan distance between them.

Task 2: Write a function called search that takes an array of Books structures and a string called title i.e. the header of the function will be: void search(struct Books b[], char title[]). This function finds the book in the array b[] whose title is the same as the parameter called title and then prints all the info (title, authors, id, subject) of that book.