**Lab Journal 14**

**Computer Programming**

**Structures**

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Enrollment #: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Class/Section: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Objective**

The objective of this lab is to assess the problem-solving capabilities of the students using the concepts acquired in the course on Programming Fundamentals.

**Task1: Give answers to the following.**

Write C++ code fragments for the following.

|  |  |
| --- | --- |
| 1. | Declare a structure Time with fields hour, minute and second. |
| struct time {  int hour;  int min;  int sec;  }; | |
| 2. | Declare a variable of type Time. |
| time s; | |
| 3. | Declare a variable of type Time, a pointer to Time and an array of 10 Times. Assign the address of the variable to the pointer. |
| time s;  time \*p;  time arr[10];  p = &s; | |
| 4. | Using pointer variable set the value of hour to 5, minute to 10 and second to 50. |
| p->hour = 5;  p->min = 10;  p->sec = 50; | |

**Task2: Implement the given exercises.**

**Exercise 1:**

Write a complete C++ program with the following features.

1. Declare a structure Point with two integer members x and y.
2. Define a function getInput(),which accepts a Point by reference. Get user input for a Point in this function.
3. Define a function addPoints()**,**which accepts two Points p1 and p2. The function adds their respective members, and returns a Point which is the sum of two. For example if one point is (2,3), the other is (4,5), the function should return a Point (6,8).
4. In the main(), declare two variables of type Point. Call the function getInput() twice to get the values of these Points from user. Add the two Points using the function addPoints() and display the x and y values of the result returned by the function.

**Code:**

#include<iostream>

#include<math.h>

using namespace std;

void getinput();

struct pointer addpoints(pointer &a, pointer &b);

struct pointer {

int x;

int y;

}p1, p2, p3;

void main()

{

getinput();

p3 = addpoints(p1, p2);

cout << p3.x << "," << p3.y << endl;;

system("pause");

}

void getinput()

{

cout << "Enter Value for x1 : ";

cin >> p1.x;

cout << "Enter Value for y1 : ";

cin >> p1.y;

cout << "Enter Value for x2 : ";

cin >> p2.x;

cout << "Enter Value for y2 : ";

cin >> p2.y;

}

pointer addpoints(pointer &a, pointer &b)

{

pointer p;

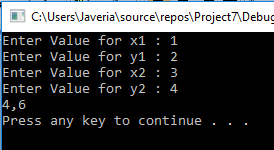
p.x = a.x + b.x;

p.y = a.y + b.y;

return p;

}

**Output:**



**Exercise 2:**

* 1. Declare a structure Rectangle with two Points as its members, the top left and the bottom right.
  2. Declare a variable of type Rectangle and get user input for the two points.
  3. Define a function computeArea() which accepts a Rectangle and returns its area.
  4. Display the area of the Rectangle in the main().

**Code:**

#include<iostream>

#include<math.h>

using namespace std;

struct rectangle {

int topleft;

int bottomright;

}p1, p2;

int computearea(rectangle a);

int main()

{

int area;

cout << "Enter hieght of the rectangle : ";

cin >> p1.topleft;

cout << "Enter width of the rectangle : ";

cin >> p1.bottomright;

area = computearea(p1);

cout << "Area of rectangle is = " << area << endl;

system("pause");

}

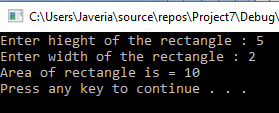
int computearea(rectangle a)

{

return a.topleft \* a.bottomright;

}

**Output:**



**Exercise 3:**

* 1. Declare a structure Time with fields hours and minutes.
  2. Declare another structure Flight with fields to store flight ID, arrival time and departure time.
  3. Define a function input(Flight \*) which allows users enter data for a flight.
  4. Define another function display(Flight \*) which displays the data for a flight.
  5. In the main, declare a variable of type Flight and call the input() and display() functions.

**Code:**

#include<iostream>

#include<math.h>

using namespace std;

struct time {

int hour;

int min;

};

struct flight {

int ID;

struct time arrivaltime;

struct time departuretime;

}p1;

void input(flight \*x);

void display(flight \*y);

int main()

{

input(&p1);

display(&p1);

system("pause");

}

void input(flight \*x)

{

cout << "Enter fight ID : ";

cin >> x->ID;

cout << "Enter depart time (hour and min) : ";

cin >> x->departuretime.hour >> x->departuretime.min;

cout << "Enter arrival time (hour and min) : ";

cin >> x->arrivaltime.hour >> x->arrivaltime.min;

cout << endl << endl;

}

void display(flight \*y)

{

cout << "Flight ID : " << y->ID << endl;

cout << "Depart Time : " << y->departuretime.hour << ":" << y->departuretime.min << endl;

cout << "Arrival Time : " << y->arrivaltime.hour << ":" << y->arrivaltime.min << endl;

}

**Output:**

