# TP
# STRUCTURES

## Exercise 1: Define and Print a Structure

1. Define a structure `Person` with the following fields:
   - `name` (string with a maximum length of 50),
   - `age` (integer),
   - `height` (float).
2. Create a `main` function where you:
   - Declare a variable of type `Person`.
   - Assign values to its fields.
   - Print the structure's values to the console.

## Exercise 2: Array of Structures

1. Define a structure `Book` with the following fields:
   - `title` (string with a maximum length of 100),
   - `author` (string with a maximum length of 50),
   - `year` (integer).
2. Create a program where you:
   - Declare an array of 3 `Book` structures.
   - Prompt the user to enter details for each book.
   - Display all the entered books in a formatted table.

## Exercise 3: Structure Comparison

1. Define a structure `Point` with the following fields:
   - `x` (integer),
   - `y` (integer).
2. Create a program where you:
   - Declare two `Point` variables.
   - Assign values to their `x` and `y` fields.
   - Write a function `int areEqual(Point p1, Point p2)` that checks if the two points are the same.
   - Use the function in `main` to print whether the points are equal or not.

## Exercise 4: Calculate Area and Perimeter

1. Define a structure `Rectangle` with the following fields:
   - `length` (float),
   - `width` (float).
2. Write a program that:
   - Declares a `Rectangle` variable.
   - Reads `length` and `width` from the user.
   - Calculates and prints the area and perimeter of the rectangle.

   Formula:
   - Area = `length * width`
   - Perimeter = `2 * (length + width)`

## Exercise 5: Structure and Functions

1. Define a structure `Student` with the following fields:
   - `id` (integer),
   - `name` (string with a maximum length of 50),
   - `grade` (float).
2. Write a program that:
   - Declares an array of 5 `Student` structures.
   - Creates a function `void inputStudent(Student* s)` to input details for a student.
   - Creates a function `void printStudent(Student s)` to print details of a student.
   - Uses these functions to input details for 5 students and display them in a formatted way.

```c
#include <stdio.h> #include <string.h>
struct Person {
    char name[50];
    int age;
    float height;
};
int main() {
    struct Person p1;
    strcpy(p1.name, "John Doe");
    p1.age = 30;
    p1.height = 5.9;
    printf("Name: %s\n", p1.name);
    printf("Age: %d\n", p1.age);
    printf("Height: %.2f\n", p1.height);
    return 0;}
```

```c
#include <stdio.h>
struct Point { int x; int y;};
int areEqual(struct Point p1, struct
Point p2) {
return (p1.x == p2.x && p1.y == p2.y);}
int main() {
    struct Point p1 = {3, 4};
    struct Point p2 = {3, 4};
    if (areEqual(p1, p2)) {
        printf("The points are
equal.\n");
    } else {
        printf("The points are not
equal.\n");
    }
    return 0;
}
```

```c
#include <stdio.h> #include <string.h>
struct Student { int id; char name[50];
    float grade;};
void inputStudent(struct Student *s) {
    printf("Enter ID: ");
    scanf("%d", &s->id);
    printf("Enter Name: ");
    scanf(" %[^\n]", s->name);
    printf("Enter Grade: ");
    scanf("%f", &s->grade);}
void printStudent(struct Student s) {
    printf("ID: %d\n", s.id);
    printf("Name: %s\n", s.name);
    printf("Grade: %.2f\n", s.grade);}
int main() {
    struct Student students[5];
    for (int i = 0; i < 5; i++) {
        printf("Enter details for student
%d:\n", i + 1);
        inputStudent(&students[i]);
    }
    printf("\nStudent Details:\n");
    for (int i = 0; i < 5; i++) {
        printf("Student %d:\n", i + 1);
        printStudent(students[i]);
    }

    return 0;}
```

```c
#include <stdio.h>
#include <string.h>
struct Book {
    char title[100];
    char author[50];
    int year;
};
int main() {
    struct Book books[3];

    // Input details
    for (int i = 0; i < 3; i++) {
        printf("Enter details for book
%d:\n", i + 1);
        printf("Title: ");
        scanf(" %[^\n]", books[i].title);
        printf("Author: ");
        scanf(" %[^\n]",
books[i].author);
        printf("Year: ");
        scanf("%d", &books[i].year);
    }
    // Display details
    printf("\nBooks Information:\n");
    for (int i = 0; i < 3; i++) {
        printf("Book %d:\n", i + 1);
        printf("Title: %s\n",
books[i].title);
        printf("Author: %s\n",
books[i].author);
        printf("Year: %d\n\n",
books[i].year);
    }
    return 0;
}
```

```c
#include <stdio.h>

struct Rectangle {
    float length;
    float width;
};

int main() {
    struct Rectangle r;

    // Input dimensions
    printf("Enter length: ");
    scanf("%f", &r.length);
    printf("Enter width: ");
    scanf("%f", &r.width);

    // Calculate and print area and
perimeter
    float area = r.length * r.width;
    float perimeter = 2 * (r.length +
r.width);

    printf("Area: %.2f\n", area);
    printf("Perimeter: %.2f\n",
perimeter);

    return 0;
}
```