Data Structures Practical Lab (0th Practical)

Name: Soniya Eknath Girhe
Section: A7 | Batch: B3
Roll No.: 59
Date of Performance: 10/01/2026

Index of Programs

1. **Arithmetic Operations (Sum, Sub, Mul, Div)**
2. **Check Positive/Negative Number**
3. **Largest of Three Numbers**
4. **Print Numbers 1–10 (Loop)**
5. **Square of a Number (Function)**
6. **Factorial (Recursive Function)**
7. **Read & Print Array Elements**
8. **Sum of Array Elements**
9. **Pointer Demonstration**
10. **Swap Numbers using Pointers**
11. **Student Information System (Structures)**
12. **Hospital Management System (Structures)**

Program 1: Arithmetic Operations

```
#include <stdio.h>
int main() {
   int a, b;
   printf("Enter two numbers: ");
```

```
    scanf("%d %d", &a, &b);

    printf("Sum = %d
", a+b);
    printf("Sub = %d
", a-b);
    printf("Mul = %d
", a*b);
    printf("Div = %d
", a/b);

    return 0;
}
```

Output Example:

```
Enter two numbers: 10 5
Sum = 15
Sub = 5
Mul = 50
Div = 2
```

Program 2: Positive or Negative Check

```
#include <stdio.h>
int main() {
    int n;
    printf("Enter a number: ");
```

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

    if (n > 0)
        printf("Positive
");
    else if (n < 0)
        printf("Negative
");
    else
        printf("Zero
");

    return 0;
}
```

Output Example:

```
Enter a number: -3
Negative
```

Program 3: Largest of Three Numbers

```
#include <stdio.h>
int main() {
    int a, b, c;
    printf("Enter three numbers: ");
    scanf("%d %d %d", &a, &b, &c);
```

```
    if (a >= b && a >= c)
        printf("Largest number is %d
", a);
    else if (b >= a && b >= c)
        printf("Largest number is %d
", b);
    else
        printf("Largest number is %d
", c);

    return 0;
}
```

Output Example:

```
Enter three numbers: 10 25 15
Largest number is 25
```

Program 4: Print Numbers 1–10 (Loop)

```
#include <stdio.h>
int main() {
    for (int i = 1; i <= 10; i++) {
        printf("%d
", i);
    }
    return 0;
}
```

Output Example:

1

2

3

4

5

6

7

8

9

10

Program 5: Square of a Number (Function)

```
#include <stdio.h>
int square(int n) {
   return n * n;
}
int main() {
   int num;
   printf("Enter a number: ");
   scanf("%d", &num);

   printf("Square of %d is %d
", num, square(num));
   return 0;
```

```
}
```

Output Example:

Enter a number: 7

Square of 7 is 49

Program 6: Factorial (Recursive Function)

```
#include <stdio.h>
long long factorial(int n) {
    if (n == 0 || n == 1) return 1;
    return n * factorial(n - 1);
}
int main() {
    int n;
    printf("Enter a number: ");
    scanf("%d", &n);

    if (n < 0)
        printf("Factorial not defined for negative numbers.
");
    else
        printf("Factorial of %d is %lld
", n, factorial(n));

    return 0;
}
```

Output Example:

Enter a number: 5

Factorial of 5 is 120

Program 7: Read & Print Array Elements

```
#include <stdio.h>
int main() {
   int n;
   printf("Enter number of elements: ");
   scanf("%d", &n);

   int arr[n];
   printf("Enter %d elements: ", n);
   for (int i = 0; i < n; i++) {
      scanf("%d", &arr[i]);
   }

   printf("Array elements are: ");
   for (int i = 0; i < n; i++) {
      printf("%d ", arr[i]);
   }
   printf("
");
   return 0;
}
```

Output Example:

Enter number of elements: 5

Enter 5 elements: 1 2 3 4 5

Array elements are: 1 2 3 4 5

Program 8: Sum of Array Elements

```
#include <stdio.h>
int main() {
   int n, sum = 0;
   printf("Enter number of elements: ");
   scanf("%d", &n);

   int arr[n];
   printf("Enter %d elements: ", n);
   for (int i = 0; i < n; i++) {
      scanf("%d", &arr[i]);
      sum += arr[i];
   }

   printf("Sum of array elements = %d
", sum);
   return 0;
}
```

Output Example:

Enter number of elements: 4

Enter 4 elements: 10 20 30 40

Sum of array elements = 100

Program 9: Pointer Demonstration

```
#include <stdio.h>
int main() {
   int var = 10;
   int *ptr = &var;

   printf("Value of var = %d
", var);
   printf("Address of var = %p
", &var);
   printf("Value stored in ptr = %p
", ptr);
   printf("Value pointed to by ptr = %d
", *ptr);

   return 0;
}
```

Output Example:

Value of var = 10

Address of var = 0x7ffee3b8a9ac

Value stored in ptr = 0x7ffee3b8a9ac

Value pointed to by ptr = 10

Program 10: Swap Numbers using Pointers

```
#include <stdio.h>
void swap(int *a, int *b) {
   int temp = *a;
   *a = *b;
   *b = temp;
}
int main() {
   int x, y;
   printf("Enter two numbers: ");
   scanf("%d %d", &x, &y);

   printf("Before swap: x = %d, y = %d
", x, y);
   swap(&x, &y);
   printf("After swap: x = %d, y = %d
", x, y);

   return 0;
}
```

Output Example:

```
Enter two numbers: 3 7
Before swap: x = 3, y = 7
After swap: x = 7, y = 3
```

Program 11: Student Information System

```
#include <stdio.h>

struct Student {
    int roll;
    char name[50];
    char course[50];
    float marks;
};

int main() {
    int n;
    printf("Enter number of students: ");
    scanf("%d", &n);
    struct Student students[n];
    for(int i = 0; i < n; i++) {
        printf("Enter details for student %d:\n", i+1);
        printf("Roll Number: ");
        scanf("%d", &students[i].roll);
        printf("Name: ");
        scanf(" %[^\n]", students[i].name);
        printf("Course: ");
        scanf(" %[^\n]", students[i].course);
        printf("Marks: ");
```

```
        scanf("%f", &students[i].marks);
    }
    printf("\nStudent Records:\n");
    printf("Roll\tName\t\tCourse\t\tMarks\n");
    for(int i = 0; i < n; i++) {
        printf("%d\t%s\t\t%s\t\t%.2f\n", students[i].roll, students[i].name,
students[i].course, students[i].marks);
    }
    return 0;
}
```

Output Example:

```
Roll   Name       Course      Marks
1      Soniya     CS          89.50
2      Rahul      IT          76.00
```

Program 12: Hospital Management System

```
#include <stdio.h>
#include <string.h>

#define MAX_PATIENTS 100

struct Patient {
    int id;
    char name[50];
    int age;
```

```
    char gender[10];
    char disease[50];
    char doctor[50];
    char admission_date[20];
};

struct Patient patients[MAX_PATIENTS];
int patient_count = 0;

void add_patient() {
    if (patient_count >= MAX_PATIENTS) {
        printf("Maximum patients reached.\n");
        return;
    }
    struct Patient p;
    printf("Enter Patient ID: ");
    scanf("%d", &p.id);
    printf("Enter Name: ");
    scanf(" %[^\n]", p.name);
    printf("Enter Age: ");
    scanf("%d", &p.age);
    printf("Enter Gender: ");
    scanf(" %[^\n]", p.gender);
    printf("Enter Disease: ");
    scanf(" %[^\n]", p.disease);
```

```
    printf("Enter Doctor Assigned: ");
    scanf(" %[^\n]", p.doctor);
    printf("Enter Admission Date (DD/MM/YYYY): ");
    scanf(" %[^\n]", p.admission_date);
    patients[patient_count++] = p;
    printf("Patient added successfully.\n");
}

void update_patient() {
    int id;
    printf("Enter Patient ID to update: ");
    scanf("%d", &id);
    for(int i = 0; i < patient_count; i++) {
        if (patients[i].id == id) {
            printf("Enter new Name: ");
            scanf(" %[^\n]", patients[i].name);
            printf("Enter new Age: ");
            scanf("%d", &patients[i].age);
            printf("Enter new Gender: ");
            scanf(" %[^\n]", patients[i].gender);
            printf("Enter new Disease: ");
            scanf(" %[^\n]", patients[i].disease);
            printf("Enter new Doctor: ");
            scanf(" %[^\n]", patients[i].doctor);
            printf("Enter new Admission Date: ");
```

```
            scanf(" %[^\n]", patients[i].admission_date);
            printf("Patient updated successfully.\n");
            return;
        }
    }
    printf("Patient not found.\n");
}

void display_all() {
    if (patient_count == 0) {
        printf("No patients to display.\n");
        return;
    }
    printf("Patient Records:\n");
    printf("ID\tName\t\tAge\tGender\tDisease\t\tDoctor\t\tAdmission
Date\n");
    for(int i = 0; i < patient_count; i++) {
        printf("%d\t%s\t\t%d\t%s\t%s\t\t%s\t\t%s\n", patients[i].id,
patients[i].name, patients[i].age, patients[i].gender, patients[i].disease,
patients[i].doctor, patients[i].admission_date);
    }
}

void display_patient() {
    int id;
    printf("Enter Patient ID to display: ");
```

```
    scanf("%d", &id);
    for(int i = 0; i < patient_count; i++) {
        if (patients[i].id == id) {
            printf("Patient Details:\n");
            printf("ID: %d\n", patients[i].id);
            printf("Name: %s\n", patients[i].name);
            printf("Age: %d\n", patients[i].age);
            printf("Gender: %s\n", patients[i].gender);
            printf("Disease: %s\n", patients[i].disease);
            printf("Doctor: %s\n", patients[i].doctor);
            printf("Admission Date: %s\n", patients[i].admission_date);
            return;
        }
    }
    printf("Patient not found.\n");
}

int main() {
    int choice;
    while(1) {
        printf("\nHospital Management System\n");
        printf("1. Add Patient\n");
        printf("2. Update Patient\n");
        printf("3. Display All Patients\n");
        printf("4. Display Specific Patient\n");
```

```
        printf("5. Exit\n");
        printf("Enter choice: ");
        scanf("%d", &choice);
        switch(choice) {
            case 1: add_patient(); break;
            case 2: update_patient(); break;
            case 3: display_all(); break;
            case 4: display_patient(); break;
            case 5: return 0;
            default: printf("Invalid choice.\n");
        }
    }
    return 0;
}
```

// Functions: add_patient, update_patient, display_all, display_patient

// Main menu-driven program

Features:

- **Add new patient records**
- **Update existing patient details**
- **Display all patients**
- **Display specific patient by ID**
