Output:

```
Enter 2 strings:
person
child
length of string: 6
Concatenated: personchild
Comparision of strings: 13
Enter substring to insert: kid
Enter position to insert substring at: 2
After substring insertion: pekidrson
Enter substring to delete: rs
Substring found at index 2!
After substring removal: peon
```

Lab Sheet 3

1. Write a program to perform the following string operations without using string handling functions:
(a) length of the string
(b) string concatenation
(c) string comparision
(d) to insert a sub string
(e) to delete a substring

Source Code:

```
#include <stdio.h>

int str_len(char* str){
  int i;
  for(i=0; str[i]; i++);
  return i;
}

int str_cmp(char* str, char* str1){
  int i=0;
  while(str[i] && str1[i]){
    if(str[i] != str1[i])
      return str[i] - str1[i];
    i++;
  }
```

Teacher's Signature _____

```
    return str[i] - str1[i];
}

int main(){
    int i, j, pos, substr_pos;
    char str[100], str1[100], substr[100];

    printf("Enter 2 strings: \n");
    scanf("%s", str);
    scanf("%s", str1);

    printf("length of str: %d\n", str_len(str));

    char str2[str_len(str) + str_len(str1)+1];
    for(i=0; str[i]; i++) str2[i] = str[i];
    for(i=0; str1[i]; i++) str2[str_len(str)+i]=str1[i];
     str2[str_len(str) + str_len(str1)] = '\0';

    printf("Concatenated: %s\n", str2);
    printf("Comparision of strings: %d\n", str_comp
(str, str1));

    printf("Enter substring to insert: ");
    scanf("%s" substr);
    printf("Enter position to insert substring at:
");
    scanf("%d", &pos);

    char str3[str_len(str) + str_len(substr)+1];
```

```
for(i=0; i<pos; i++) str3[i] = str[i];
for(i=0; substr[i]; i++) str3[pos+i] = substr[i];
for(i=pos; str[i]; i++) str3[str_len(substr)+i] =
str[i];
str3[str_len(str) + str_len(substr)] = '\0';

printf("After substring insertion: %s\n", str3);

printf("Enter substring to delete: ");
scanf("%s", substr);

substr_pos = -1;

for(i=0; str[i]; i++) {
  int matching = 1;

  for(j=0; substr[j]; j++)
    if(substr[j] != str[i+j]) matching = 0;

  if(matching == 1) {
    substr_pos = i;
    break;
  }
}

if(substr_pos == -1) {
  printf("Substring not found in string\n");
  return 1;
}
```

Teacher's Signature _____

```
    printf("Substring found at index %d!\n", sub-
-str_pos);

    for(i=0; substr[i]; i++)
        for(j=substr_pos; str[j]; j++)
            str[j]=str[j+1];

    printf("After substring removal: %s\n", str);

    return 0;
}
```

Teacher's Signature _____

Output:

```
Enter number of students in school: 3
Enter roll number of student: 3
Enter name of student: Jerome
Enter grade of student: A
Enter roll number of student: 2
Enter name of student: Kalp
Enter grade of student: A
Enter roll number of student: 1
Enter name of student: Pranav
Enter grade of student: A
```

Name	Roll no.	Grade
Pranav	1	A
Kalp	2	A
Jerome	3	A

2. Write a C program to define a student structure with the data members to store name, roll no. & grade of the student. Also write the required functions to read, display, and sort student information according to the roll number of the student. All the member functions will have an array of objects as arguments.

Source Code:

```
#include <stdio.h>

typedef struct {
    char name[50];
    int roll_no;
    char grade;
} Student;

void read(Student * students, int n) {
    int i;
    for(i=0; i<n; i++) {
        printf("Enter roll number of student: ");
        scanf("%d", &students[i].roll_no);
        printf("Enter name of student: ");
        scanf("%s", students[i].name);
        getchar();
        printf("Enter grade of student ");
        scanf("%c", &students[i].grade);
```

Teacher's Signature _____

Date

Expt. No. Page No.

```
    }
  }

void sort(Student * students, int n) {
    int i, j;
    for(i=0; i<n-1; i++) {
        for(j=0; j<n-i-1; j++) {
            if(students[j].roll-no > students[j+1].roll-
no) {
                Student temp = students[j];
                students[j] = students[j+1];
                students[j+1] = temp;
            }
        }
    }
}

void display(Student * students, int n) {
    int i;
    printf("Name \t \t Roll no. \t \t Grade \n");
    for(i=0; i<n; i++)
        printf("%s \t \t %d \t \t %c \n", students[i]
.name, students[i].roll-no, students[i].grade);
}

int main() {
    int n;
    printf("Enter number of students in school:");
    scanf("%d", &n);
```

Teacher's Signature

```
    Student students[n];

    read(students, n);
    sort(students, n);
    display(students, n);

    return 0;
}
```

Teacher's Signature _____

Output:

```
Enter number of students in school: 3
Enter roll number of student : 3
Enter name of student: Rahul
Enter marks of student: 50
Enter roll number of student: 2
Enter name of student: Vaibhav
Enter marks of student: 70
Enter roll number of student: 1
Enter name of student: XYZ
Enter marks of student: 60
Name        Roll no.        Marks
Rahul       3               50.000000
Vaibhav     2               70.000000
XYZ         1               60.000000
Details of students with highest marks:
Name        Roll no.        Marks
Vaibhav     2               70.000000
```

3. Define a structure Student with the following members:
char name[50] - to store student name as string
int roll_no - to store roll no.
float marks - to store marks
Write a C program that:
(i) Reads the details of 'n' students using a function that uses pointer to structure as an argument.
(ii) Displays the details of all students using a separate function.
(iii) Finds and displays the student with the highest marks using pointer-based access.

Source Code:

```
#include <stdio.h>

typedef struct {
    char name[50];
    int roll_no;
    float marks;
} Student;

void read(Student * students, int n) {
    int i;
    for(i=0; i<n; i++) {
        printf("Enter roll number of student: ");
```

Teacher's Signature _____

```
        scanf("%d", &(students+i)->roll_no);
        printf("Enter name of student: ");
        scanf("%s", &(students+i)->name);
        getchar();
        printf("Enter marks of student: ");
        scanf("%f", &(students+i)->marks);
    }
}

void display(Student* students, int n) {
    int i;
    printf("Name \t\t Roll no. \t\t Marks \n");
    for(i=0; i<n; i++)
        printf("%s \t\t %d \t\t %f \n", (students+i)
-> name, (students+i)->roll_no, (students+i)->
marks);
}

void highest(Student* students, int n) {
    Student high;
    high.marks = -1;
    int i;
    for(i=0; i<n; i++) {
        if((students+i)->marks > high.marks){
            high = *(students+i);
        }
    }
    Student highArr[] = {high};
    printf("Details of students with highest marks:\n");
```

Teacher's Signature _____

```
    display(highArr, 1);
}

int main() {
    int n;
    printf("Enter number of students in sch-
-ool: ");
    scanf("%d", &n);

    Student students[n];

    read(students, n);
    display(students, n);
    highest(students, n);

    return 0;
}
```

Teacher's Signature _____