



JAMIA MILLIA ISLAMIA

C LAB PROGRAMS

IIIRD SEMESTER

SESSION: 2022-2023

- MD. IBRAHIM AKHTAR
- Dept. of Computer Science
- Roll No.: 21BCS007
- Serial No.: 07
- Semester: 3

INDEX

Sr. No.:	Date:	Program:	Remark	Signature
1	18/7/2022	Menu Driven for Searching Algorithms		
2	25/7/2022	Menu Driven for Sorting Algorithms		
3	1/8/2022	Matrix Addition, Multiplication, and Transpose		
4	22/8/2022	Conversion from Decimal to Octal and vice versa		
5	29/8/2022	Largest Sum Contiguous Array and its Length		
6	12/9/2022	Implement user-defined String Library Functions		
7	26/9/2022	Store Student details using 2D Array		
8	17/10/2022	Text Wrap to replace a Substring with another String		
9	14/11/2022	Encoding and Decoding Mechanism		
10	14/11/2022	Complex Numbers operations using Structures		
11	14/11/2022	Sum and Deviation of elements stored in an Array		
12	14/11/2022	Difference b/w 2 Dates		
13	14/11/2022	Menu Driven to implement File Handling		
14	14/11/2022	Complex Numbers operations using Strings		

Program: 1

C ibrahim_lab1.c >  linearSearch(int [], int, int)

```
1  // Binary & Linear Search
2
3  #include <stdio.h>
4
5  int linearSearch(int array[], int n, int x) // Linear Search
6  {
7      for (int i = 0; i < n; i++)
8          if (array[i] == x)
9              return i;
10     return -1;
11 }
12
13 int binarySearch(int array[], int n, int x) // Binary Search
14 {
15     int low = 0;
16     int high = n - 1;
17     while (low <= high)
18     {
19         int mid = low + (high - low) / 2;
20         if (array[mid] == x)
21             return mid;
22         if (array[mid] < x)
23             low = mid + 1;
24         else
25             high = mid - 1;
26     }
27     return -1;
28 }
29
30 int main()
```

C ibrahim_lab1.c > linearSearch(int [], int, int)

```
31 {
32     printf("Enter the size of array you want to create: ");
33     int size;
34     scanf("%d", &size);
35     int a[size];
36     printf("\nEnter %d elements in array.\n", size);
37     for (int i = 0; i < size; i++)
38     {
39         printf("Enter %d element: ", i + 1);
40         scanf("%d", &a[i]);
41     }
42     while (1)
43     {
44         printf("\nMENU:\n1. For Linear Searching\n2. For Binary Searching\n3. Display all Elements\n4. Exit: ");
45         int choice;
46         scanf("%d", &choice);
47         switch (choice)
48         {
49             case 1:
50             {
51                 printf("Enter the number you want to search: ");
52                 int num;
53                 scanf("%d", &num);
54                 int index = linearSearch(a, size, num);
55                 if (index == -1)
56                     printf("Number %d you Searched was not found in the array!!!\n", num);
57                 else
58                     printf("Number %d you Searched was found in the array at index %d\n", num, index + 1);
59                 break;
60             }

```

C ibrahim_lab1.c > linearSearch(int [], int, int)

```
61         case 2:
62         {
63             printf("Enter the number you want to search: ");
64             int num;
65             scanf("%d", &num);
66             int index = binarySearch(a, size, num);
67             if (index == -1)
68                 printf("Number %d you Searched was not found in the array!!!\n", num);
69             else
70                 printf("Number %d you Searched was found in the array at index %d\n", num, index + 1);
71             break;
72         }
73         case 3:
74         {
75             for (int i = 0; i < size; i++)
76             {
77                 printf("Element present at position %d is %d\n", i + 1, a[i]);
78             }
79             break;
80         }
81         case 4:
82             return 0;
83             break;
84         default:
85             printf("Wrong Input\n");
86             break;
87     }
88 }
89 return 0;
90 }
```

Output: 1

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab1.c
PS D:\Programming\C Lab Programs HW> .\a.exe
Enter the size of array you want to create: 7
```

```
Enter 7 elements in array.
Enter 1 element: -1
Enter 2 element: 7
Enter 3 element: 18
Enter 4 element: 25
Enter 5 element: 50
Enter 6 element: 81
Enter 7 element: 100
```

```
MENU:
1. For Linear Searching
2. For Binary Searching
3. Display all Elements
4. Exit: 3
Element present at position 1 is -1
Element present at position 2 is 7
Element present at position 3 is 18
Element present at position 4 is 25
Element present at position 5 is 50
Element present at position 6 is 81
Element present at position 7 is 100
```

```
MENU:
1. For Linear Searching
2. For Binary Searching
3. Display all Elements
4. Exit: 1
Enter the number you want to search: 50
Number 50 you Searched was found in the array at index 5
```

```
MENU:
1. For Linear Searching
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
4. Exit: 1
Enter the number you want to search: 50
Number 50 you Searched was found in the array at index 5
```

```
MENU:
1. For Linear Searching
2. For Binary Searching
3. Display all Elements
4. Exit: 2
Enter the number you want to search: 100
Number 100 you Searched was found in the array at index 7
```

```
MENU:
1. For Linear Searching
2. For Binary Searching
3. Display all Elements
4. Exit: 4
PS D:\Programming\C Lab Programs HW> █
```

Program: 2

```
1 // Sorting in ascending and descending order
2
3 #include <stdio.h>
4
5 void sortAscending(int a[], int size)
6 {
7     for (int i = 0; i < size; ++i)
8     {
9         for (int j = i + 1; j < size; ++j)
10         {
11             if (a[i] > a[j])
12             {
13                 int x = a[i];
14                 a[i] = a[j];
15                 a[j] = x;
16             }
17         }
18     }
19 }
20
21 void sortDescending(int a[], int size)
22 {
23     for (int i = 0; i < size; ++i)
24     {
25         for (int j = i + 1; j < size; ++j)
26         {
27             if (a[i] < a[j])
28             {
29                 int x = a[i];
30                 a[i] = a[j];
```

C ibrahim_lab2.c X

C ibrahim_lab2.c > ...

```
31         a[j] = x;
32     }
33 }
34 }
35 }
36
37 int main()
38 {
39     printf("Enter the size of array you want to create: ");
40     int size;
41     scanf("%d", &size);
42     int a[size];
43     printf("\nEnter %d elements in array.\n", size);
44     for (int i = 0; i < size; i++)
45     {
46         printf("Enter %d element: ", i + 1);
47         scanf("%d", &a[i]);
48     }
49     while (1)
50     {
51         printf("\nMENU:\n1. For Ascending Order\n2. For Descending Order\n3. Display all Elements\n4. Exit: ");
52         int choice;
53         scanf("%d", &choice);
54         switch (choice)
55         {
56             case 1:
57             {
58                 sortAscending(a, size);
59                 printf("The array you entered has been Sorted in Ascending Order\n");
60                 break;
```

C ibrahim_lab2.c X

C ibrahim_lab2.c > ...

```
61     }
62     case 2:
63     {
64         sortDescending(a, size);
65         printf("The array you entered has been Sorted in Descending Order\n");
66         break;
67     }
68     case 3:
69     {
70         printf("Displaying Array:\n");
71         for (int i = 0; i < size; i++)
72         {
73             printf("Element present at position %d is %d\n", i + 1, a[i]);
74         }
75         break;
76     }
77     case 4:
78         return 0;
79         break;
80     default:
81         printf("Invalid Input!!!\n");
82         break;
83     }
84 }
85 }
```

Output: 2

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

```
PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab2.c
```

```
PS D:\Programming\C Lab Programs HW> .\a.exe
```

Enter the size of array you want to create: 7

Enter 7 elements in array.

Enter 1 element: 7

Enter 2 element: -1

Enter 3 element: 34

Enter 4 element: 90

Enter 5 element: 56

Enter 6 element: 2

Enter 7 element: 81

MENU:

1. For Ascending Order

2. For Descending Order

3. Display all Elements

4. Exit: 1

The array you entered has been Sorted in Ascending Order

MENU:

1. For Ascending Order

2. For Descending Order

3. Display all Elements

4. Exit: 3

Displaying Array:

Element present at position 1 is -1

Element present at position 2 is 2

Element present at position 3 is 7

Element present at position 4 is 34

Element present at position 5 is 56

Element present at position 6 is 81

Element present at position 7 is 90

MENU:

1. For Ascending Order

2. For Descending Order

3. Display all Elements

4. Exit: 2

The array you entered has been Sorted in Descending Order

MENU:

1. For Ascending Order

2. For Descending Order

3. Display all Elements

4. Exit: 3

Displaying Array:

Element present at position 1 is 90

Element present at position 2 is 81

Element present at position 3 is 56

Element present at position 4 is 34

Element present at position 5 is 7

Element present at position 6 is 2

Element present at position 7 is -1

MENU:

1. For Ascending Order

2. For Descending Order

3. Display all Elements

4. Exit: 4

PS D:\Programming\C Lab Programs HW> █

Program: 3

C ibrahim_lab3.c ×

C ibrahim_lab3.c > add(int, int, int [row][col], int [row][col])

```
1 // matrix
2
3 #include <stdio.h>
4
5 void add(int row, int col, int a[row][col], int b[row][col])
6 {
7     int c[row][col];
8     for (int i = 0; i < row; i++)
9     {
10         for (int j = 0; j < col; j++)
11         {
12             c[i][j] = a[i][j] + b[i][j];
13         }
14     }
15     printf("After addition, Result Matrix:\n");
16     for (int i = 0; i < row; i++)
17     {
18         for (int j = 0; j < col; j++)
19             printf("%d ", c[i][j]);
20         printf("\n");
21     }
22 }
23
24 void multiply(int row, int col, int a[row][col], int b[row][col])
25 {
26     int c[row][col];
27     for (int i = 0; i < row; i++)
28     {
29         for (int j = 0; j < col; j++)
30         {
```

C ibrahim_lab3.c ×

C ibrahim_lab3.c > add(int, int, int [row][col], int [row][col])

```
31         int temp = 0;
32         for (int k = 0; k < row; k++)
33         {
34             temp = temp + (a[i][k] * b[k][j]);
35             c[i][j] = temp;
36         }
37     }
38 }
39 printf("After multiplication, Result Matrix:\n");
40 for (int i = 0; i < row; i++)
41 {
42     for (int j = 0; j < col; j++)
43         printf("%d ", c[i][j]);
44     printf("\n");
45 }
46 }
47
48 void transpose(int row, int col, int a[row][col])
49 {
50     int c[row][col];
51     for (int i = 0; i < row; i++)
52     {
53         for (int j = 0; j < col; j++)
54         {
55             c[i][j] = a[j][i];
56         }
57     }
58     printf("After transpose, Result Matrix:\n");
59     for (int i = 0; i < row; i++)
60     {
```

C ibrahim_lab3.c ×

```
C ibrahim_lab3.c > add(int, int, int [row][col], int [row][col])
61     for (int j = 0; j < col; j++)
62     |     printf("%d ", c[i][j]);
63     |     printf("\n");
64     | }
65     | }
66
67     int main()
68     {
69         printf("Enter Row Size of 2D Matrix: ");
70         int row;
71         scanf("%d", &row);
72         printf("Enter Column Size of 2D Matrix: ");
73         int col;
74         scanf("%d", &col);
75
76         int a[row][col], b[row][col];
77
78         printf("For 1st Array\n");
79         for (int i = 0; i < row; i++)
80         {
81             for (int j = 0; j < row; j++)
82             {
83                 printf("Enter element of a[%d][%d]: ", i + 1, j + 1);
84                 scanf("%d", &a[i][j]);
85             }
86         }
87
88         printf("For 2nd Array\n");
89         for (int i = 0; i < row; i++)
90         {
```

C ibrahim_lab3.c ×

```
C ibrahim_lab3.c > add(int, int, int [row][col], int [row][col])
91     for (int j = 0; j < row; j++)
92     {
93         printf("Enter element of b[%d][%d]: ", i + 1, j + 1);
94         scanf("%d", &b[i][j]);
95     }
96 }
97
98 while (1)
99 {
100     printf("\nMENU:\n1. Addition\n2. Multiplication\n3. Transpose\n4. Exit: ");
101     int choice;
102     scanf("%d", &choice);
103
104     switch (choice)
105     {
106     case 1:
107     {
108         add(row, col, a, b);
109         break;
110     }
111     case 2:
112     {
113         multiply(row, col, a, b);
114         break;
115     }
116     case 3:
117     {
118         printf("Transpose which matrix?\nEnter 1 for 1st Matrix\nEnter 2 for 2nd Matrix: \n");
119         int x;
120         scanf("%d", &x);
121         if (x == 1)
122         |     transpose(row, col, a);
123         |     else if (x == 2)
124         |     transpose(row, col, b);
125         |     else
126         |     printf("Invalid Input, Try Again!!!\n");
127         |     break;
128         }
129     case 4:
130         return 0;
131         break;
132     default:
133         printf("Invalid Input!!!\n");
134         break;
135     }
136 }
137 return 0;
138 }
```

Output: 3

```
PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab3.c
PS D:\Programming\C Lab Programs HW> .\a.exe
Enter Row Size of 2D Matrix: 2
Enter Column Size of 2D Matrix: 2
For 1st Array
Enter element of a[1][1]: 24
Enter element of a[1][2]: 7
Enter element of a[2][1]: 17
Enter element of a[2][2]: 9
For 2nd Array
Enter element of b[1][1]: 1
Enter element of b[1][2]: 15
Enter element of b[2][1]: 13
Enter element of b[2][2]: 7

MENU:
1. Addition
2. Multiplication
3. Transpose
4. Exit: 1
After addition, Result Matrix:
25 22
30 16

MENU:
1. Addition
2. Multiplication
3. Transpose
4. Exit: 2
After multiplication, Result Matrix:
115 409
134 318

MENU:
1. Addition
2. Multiplication
3. Transpose
4. Exit: 3
Transpose which matrix?
Enter 1 for 1st Matrix
Enter 2 for 2nd Matrix:
1
After transpose, Result Matrix:
24 17
7 9

MENU:
1. Addition
2. Multiplication
3. Transpose
4. Exit: 3
Transpose which matrix?
Enter 1 for 1st Matrix
Enter 2 for 2nd Matrix:
2
After transpose, Result Matrix:
1 13
15 7

MENU:
1. Addition
2. Multiplication
3. Transpose
4. Exit: 4
PS D:\Programming\C Lab Programs HW> █
```

Program: 4

C ibrahim_lab4.c > ...

```
1 //Decimal to Octal & Octal to Decimal Conversion
2
3 #include<stdio.h>
4 #include <stdlib.h>
5 #include <string.h>
6
7 char *decimalToOctal(int decimal)
8 {
9     char *octal = (char *)malloc(23);
10    if (octal == NULL)
11    {
12        printf("Clear Memory!!!\n");
13        return 0;
14    }
15    octal = octal + 22;
16    *octal-- = '\0';
17    if (decimal == 0)
18    {
19        *octal = '0';
20    }
21    else
22    {
23        char remainder;
24        while (decimal > 0)
25        {
26            remainder = (decimal % 8) + '0';
27            *octal-- = remainder;
28            decimal = decimal / 8;
29        }
30        octal++;
31    }
32    return octal;
33 }
```

C ibrahim_lab4.c > ...

```
31 }
32
33 int octToDecimal(char *oct, int length)
34 {
35     int decimal = 0;
36     int x = 1;
37     oct = oct + length - 1;
38     for (int i = 0; i < length; i++, oct--)
39     {
40         int coefficient = *oct - '0';
41         decimal = decimal + (x * coefficient);
42         x = x * 8;
43     }
44
45     return decimal;
46 }
47
48 int main()
49 {
50     while (1)
51     {
52         printf("\nMENU:\n1. Decimal to Octal\n2. Octal to Decimal\n3. Exit: ");
53         int choice;
54         scanf("%d", &choice);
55
56         switch (choice)
57         {
58             case 1:
59             {
60                 int decimal;
```

```

C ibrahim_lab4.c > main()
61     char *octal;
62     printf("Enter the Decimal Number you want to convert: ");
63     scanf("%d", &decimal);
64     octal = decimalToOctal(decimal);
65     printf("%d in Decimal equals %s in Octal\n", decimal, octal);
66     break;
67 }
68 case 2:
69 {
70     char oct[23];
71     int length;
72     int decimal;
73     printf("Enter the Octal Number you want to convert: ");
74     scanf("\n%22s", oct);
75     length = strlen(oct);
76     decimal = octToDecimal(oct, length);
77     printf("%s in Octal is %d in Decimal\n", oct, decimal);
78     break;
79 }
80 case 3:
81     return 0;
82     break;
83 default:
84     printf("Invalid Input!!!\n");
85     break;
86 }
87 }
88 return 0;
89 }

```

Output: 4

```

PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab4.c
PS D:\Programming\C Lab Programs HW> .\a.exe

```

MENU:

1. Decimal to Octal
2. Octal to Decimal
3. Exit: 1

```

Enter the Decimal Number you want to convert: 19
19 in Decimal equals 23 in Octal

```

MENU:

1. Decimal to Octal
2. Octal to Decimal
3. Exit: 2

```

Enter the Octal Number you want to convert: 23
23 in Octal is 19 in Decimal

```

MENU:

1. Decimal to Octal
2. Octal to Decimal
3. Exit: 3

```

PS D:\Programming\C Lab Programs HW> █

```

Program: 5

C ibrahim_lab5.c ×

C ibrahim_lab5.c > ...

```
1  // largest contiguous subarray
2
3  #include <stdio.h>
4
5  int main()
6  {
7      int size, m = 0, l = 0;
8      printf("Enter the size of the array\n");
9      scanf("%d", &size);
10     int array[size];
11     printf("Enter the Elements of the array\n");
12
13     for (int i = 0; i < size; i++)
14     {
15         scanf("%d", &array[i]);
16     }
17     int max = array[0];
18     for (int i = 0; i < size; i++)
19     {
20         int sum = 0;
21         for (int j = i; j < size; j++)
22         {
23             sum = sum + array[j];
24             if (sum > max)
25             {
26                 m = i;
27                 l = j;
28                 max = sum;
29             }
30         }
```

C ibrahim_lab5.c ×

C ibrahim_lab5.c > main()

```
31     }
32     printf("\nThe length of max contiguous subarray is %d\nElements are: ", l - m + 1);
33     for (int i = m; i <= l; i++)
34     {
35         printf("%d\t", array[i]);
36     }
37     printf("\nThe sum of the max contiguous subarray is %d\n", max);
38     return 0;
39 }
```

Output: 5

```
PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab5.c
PS D:\Programming\C Lab Programs HW> .\a.exe
Enter the size of the array
7
Enter the Elements of the array
-1
6
8
-15
7
9
0

The length of max contiguous subarray is 2
Elements are: 7 9
The sum of the max contiguous subarray is 16
PS D:\Programming\C Lab Programs HW> 
```

Program: 6

C ibrahim_lab6.c X

C ibrahim_lab6.c > ...

```

1 // making library functions of string
2
3 #include <stdio.h>
4 #include <string.h>
5 #include <stdlib.h>
6 int main()
7 {
8     char str1[20], str2[20];
9     int i, j;
10
11     while (1)
12     {
13         printf("\nMENU\n1:Find Length of String\n2:Find Reverse of String\n3:Copy String\n4:Compare Strings");
14         printf("\n5:Concatenate Strings\n6:Check for Palindrome\n7:Find substring\n8:Exit : ");
15         int ch;
16         scanf("%d", &ch);
17         switch (ch)
18         {
19             case 1:
20                 printf("Enter String: ");
21                 scanf("%s", str1);
22                 i = strlen(str1);
23                 printf("Length of String : %d\n", i);
24                 break;
25
26             case 2:
27                 printf("Enter String: ");
28                 scanf("%s", str1);
29                 strrev(str1);
30                 printf("Reverse string : %s\n", str1);

```

```

31         break;
32
33     case 3:
34         printf("Enter String1: ");
35         scanf("%s", str1);
36         printf("Enter String2: ");
37         scanf("%s", str2);
38         printf("String Before Copying:\nString1=\"%s\" , String2=\"%s\"\n", str1, str2);
39         strcpy(str2, str1);
40         printf("String After Copying:\nString1=\"%s\" , String2=\"%s\"\n", str1, str2);
41         break;
42
43     case 4:
44         printf("Enter First String: ");
45         scanf("%s", str1);
46         printf("Enter Second String: ");
47         scanf("%s", str2);
48         j = strcmp(str1, str2);
49         if (j == 0)
50             printf("Strings are Same\n");
51         else
52             printf("Strings are Not Same\n");
53         break;
54
55     case 5:
56         printf("\nEnter First String: ");
57         scanf("%s", str1);
58         printf("Enter Second string: ");
59         scanf("%s", str2);
60         strcat(str1, str2);
61
62         printf("String After Concatenation : %s\n", str1);
63         break;
64
65     case 6:
66         printf("Enter String:");
67         scanf("%s", &str1);
68         int l = 0;
69         int h = strlen(str1) - 1;
70         int x = 0;
71         while (h > l)
72         {
73             if (str1[l++] != str1[h--])
74             {
75                 printf("%s is not a palindrome\n", str1);
76                 x = 1;
77                 break;
78             }
79         }
80         if (x == 0)
81             printf("%s is a palindrome\n", str1);
82         break;
83
84     case 7:
85         printf("Enter First String:");
86         scanf("%s", str1);
87         printf("Enter Second String:");
88         scanf("%s", str2);
89         if (strstr(str1, str2) == NULL)
90             printf("Second String is Not a Substring of First String\n");
91         else

```



```
C ibrahim_lab6.c ×
C ibrahim_lab6.c > main()

91 |         printf("Second String is a Substring of First String\n");
92 |         break;
93 |
94 |     case 8:
95 |         exit(0);
96 |         break;
97 |     default:
98 |         printf("Invalid Input!!!\n");
99 |     }
100 | }
101 | return 0;
102 | }
```

Output: 6

```
PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab6.c
PS D:\Programming\C Lab Programs HW> .\a.exe
```

MENU

```
1:Find Length of String
2:Find Reverse of String
3:Copy String
4:Compare Strings
5:Concatenate Strings
6:Check for Palindrome
7:Find substring
8:Exit : 1
Enter String: ibrahim
Length of String : 7
```

MENU

```
1:Find Length of String
2:Find Reverse of String
3:Copy String
4:Compare Strings
5:Concatenate Strings
6:Check for Palindrome
7:Find substring
8:Exit : 2
Enter String: ibrahim
Reverse string : miharbi
```

MENU

```
1:Find Length of String
2:Find Reverse of String
3:Copy String
4:Compare Strings
5:Concatenate Strings
6:Check for Palindrome
7:Find substring
8:Exit : 3
```

```
Enter String1: ibrahim
Enter String2: akhtar
String Before Copying:
String1="ibrahim" , String2="akhtar"
String After Copying:
String1="ibrahim" , String2="ibrahim"
```

MENU

```
1:Find Length of String
2:Find Reverse of String
3:Copy String
4:Compare Strings
5:Concatenate Strings
6:Check for Palindrome
7:Find substring
8:Exit : 4
Enter First String: ibrahim
Enter Second String: ibrahim
Strings are Same
```

MENU

```
1:Find Length of String
2:Find Reverse of String
3:Copy String
4:Compare Strings
5:Concatenate Strings
6:Check for Palindrome
7:Find substring
8:Exit : 5
```

```
Enter First String: md-ibrahim
Enter Second string: akhtar
String After Concatenation : md-ibrahimakhtar
```

MENU

```
1:Find Length of String
2:Find Reverse of String
3:Copy String
4:Compare Strings
5:Concatenate Strings
6:Check for Palindrome
7:Find substring
8:Exit : 6
Enter String:ibrahim
ibrahim is not a palindrome
```

MENU

```
1:Find Length of String
2:Find Reverse of String
3:Copy String
4:Compare Strings
5:Concatenate Strings
6:Check for Palindrome
7:Find substring
8:Exit : 7
Enter First String:ibrahim
Enter Second String:md
Second String is Not a Substring of First String
```

MENU

```
1:Find Length of String
2:Find Reverse of String
3:Copy String
4:Compare Strings
5:Concatenate Strings
6:Check for Palindrome
7:Find substring
8:Exit : 8
```

PS D:\Programming\C Lab Programs HW> █

Program: 7

```
C ibrahim_lab7.c > main()
1 // 2D array (for data of students)
2
3 #include <stdio.h>
4 #include <stdlib.h>
5 int main()
6 {
7     printf("Enter the Number of Students: ");
8     int n;
9     scanf("%d", &n);
10    int M[n][6];
11
12    int i = 0, j = 0;
13    printf("Enter the Roll Number of Students\n");
14    for (j = 0; i < n; i++)
15    {
16        scanf("%d", &M[i][j]);
17    }
18    printf("Enter the Age of Students\n");
19    i = 0;
20    for (j = 1; i < n; i++)
21    {
22        scanf("%d", &M[i][j]);
23    }
24    printf("Enter the marks of Subject 1\n");
25    i = 0;
26    for (j = 2; i < n; i++)
27    {
28        scanf("%d", &M[i][j]);
29    }
30    printf("Enter the marks of Subject 2\n");
31
32    i = 0;
33    for (j = 3; i < n; i++)
34    {
35        scanf("%d", &M[i][j]);
36    }
37    printf("Enter the marks of Subject 3\n");
38    i = 0;
39    for (j = 4; i < n; i++)
40    {
41        scanf("%d", &M[i][j]);
42    }
43    printf("Enter the marks of Subject 4\n");
44    i = 0;
45    for (j = 5; i < n; i++)
46    {
47        scanf("%d", &M[i][j]);
48    }
49    printf("Roll_no\tAge\tSubject 1\tSubject 2\tSubject 3\tSubject 4\n");
50    for (i = 0; i < n; i++)
51    {
52        for (j = 0; j < 6; j++)
53        {
54            printf("%d\t", M[i][j]);
55        }
56        printf("\n");
57    }
58    printf("\n");
59    printf("Percentage with Roll Numbers\n");
60    for (i = 0; i < n; i++)
```

C ibrahim_lab7.c > main()

```
61 {
62     float sum = 0;
63     printf("%d - ", M[i][0]);
64     for (j = 2; j < 6; j++)
65     {
66         sum += M[i][j];
67     }
68     printf("%.3f\n", (sum) / 4);
69 }
70 printf("\n");
71 printf("\n");
72 printf("Highest marks in each subject along with Roll Number\n");
73 for (j = 2; j < 6; j++)
74 {
75     int a = 0;
76     int max = 0;
77     for (i = 0; i < n; i++)
78     {
79         if (M[i][j] == max)
80         {
81             if (M[i][1] <= M[a][1])
82             {
83                 a = i;
84                 max = M[i][j];
85             }
86         }
87         if (M[i][j] > max)
88         {
89             a = i;
90             max = M[i][j];
91         }
92     }
93     printf("subject %d-", (j - 1));
94     printf(" %d by roll number %d\n", max, M[a][0]);
95 }
96 printf("\n");
97 printf("\n");
98 printf("Student with Highest Percentage\n");
99 float final = 0;
100 int a = 0;
101 for (i = 0; i < n; i++)
102 {
103     float percent = 0;
104     for (j = 2; j < 6; j++)
105     {
106         percent += M[i][j];
107     }
108     if ((percent / 4) >= final)
109     {
110         if ((percent / 4) == final)
111         {
112             if (M[i][1] < M[a][1])
113             {
114                 a = i;
115                 final = (percent / 4);
116             }
117         }
118         else
119         {
120             final = (percent / 4);
121             a = i;
122         }
123     }
124 }
125 printf("Highest Percentage is %f by Roll Number %d\n", final, M[a][0]);
126 printf("\n");
127 printf("\n");
128 return 0;
129 }
```

C ibrahim_lab7.c > main()

```
91 }
92 }
93 printf("subject %d-", (j - 1));
94 printf(" %d by roll number %d\n", max, M[a][0]);
95 }
96 printf("\n");
97 printf("\n");
98 printf("Student with Highest Percentage\n");
99 float final = 0;
100 int a = 0;
101 for (i = 0; i < n; i++)
102 {
103     float percent = 0;
104     for (j = 2; j < 6; j++)
105     {
106         percent += M[i][j];
107     }
108     if ((percent / 4) >= final)
109     {
110         if ((percent / 4) == final)
111         {
112             if (M[i][1] < M[a][1])
113             {
114                 a = i;
115                 final = (percent / 4);
116             }
117         }
118         else
119         {
120             final = (percent / 4);
121             a = i;
122         }
123     }
124 }
125 printf("Highest Percentage is %f by Roll Number %d\n", final, M[a][0]);
126 printf("\n");
127 printf("\n");
128 return 0;
129 }
```

Output: 7

```
PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab7.c
PS D:\Programming\C Lab Programs HW> .\a.exe
Enter the Number of Students: 3
Enter the Roll Number of Students
7
1
2
Enter the Age of Students
50
51
52
Enter the marks of Subject 1
70
95
100
Enter the marks of Subject 2
25
15
10
Enter the marks of Subject 3
50
50
50
Enter the marks of Subject 4
35
25
45
Roll_no Age      Subject 1      Subject 2      Subject 3      Subject 4
7       50       70       25       50       35
1       51       95       15       50       25
2       52       100      10       50       45
Percentage with Roll Numbers
7 - 45.000
1 - 46.250
2 - 51.250

Highest marks in each subject along with Roll Number
subject 1- 100 by roll number 2
subject 2- 25 by roll number 7
subject 3- 50 by roll number 7
subject 4- 45 by roll number 2

Student with Highest Percentage
Highest Percentage is 51.250000 by Roll Number 2

PS D:\Programming\C Lab Programs HW> █
```

Program: 8

```
C ibrahim_lab8.c > main()
1  // replace substring
2  #include <stdio.h>
3  #include <string.h>
4  int main()
5  {
6      char str[256], substr[128], replace[128], output[256];
7      printf("Enter the string: ");
8      scanf("%s", &str);
9      printf("Enter the substring: ");
10     scanf("%s", &substr);
11     printf("Enter the string you want substring to get replaced with: ");
12     scanf("%s", &replace);
13     int i = 0, j = 0, flag = 0, start = 0;
14     str[strlen(str)] = '\0';
15     substr[strlen(substr)] = '\0';
16     replace[strlen(replace)] = '\0';
17     while (str[i] != '\0') // checking whether the substring to be replaced is present
18     {
19         if (str[i] == substr[j])
20         {
21             if (!flag)
22                 start = i;
23             j++;
24             if (substr[j] == '\0')
25                 break;
26             flag = 1;
27         }
28         else
29         {
30             flag = start = i = 0;
```

```

C ibrahim_lab8.c > main()
31     }
32     i++;
33 }
34 if (substr[j] == '\0' && flag)
35 {
36     for (i = 0; i < start; i++)
37         output[i] = str[i];
38
39     for (j = 0; j < strlen(replace); j++) // replace substring with another string
40     {
41         output[i] = replace[j];
42         i++;
43     }
44
45     for (j = start + strlen(substr); j < strlen(str); j++) // copying remaining portion of the input string
46     {
47         output[i] = str[j];
48         i++;
49     }
50
51     output[i] = '\0';
52     printf("Output: %s\n", output);
53 }
54 else
55 {
56     printf("%s is not a substring of %s\n", substr, str);
57 }
58 return 0;
59 }

```

Output: 8

```

PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab8.c
PS D:\Programming\C Lab Programs HW> .\a.exe
Enter the string: md-ibrahim
Enter the substring: ibrahim
Enter the string you want substring to get replaced with: akhtar
Output: md-akhtar
PS D:\Programming\C Lab Programs HW> █

```

Program: 9

```

C ibrahim_lab9.c > ...
1 //encoding and decoding
2
3 #include <stdio.h>
4 #include <string.h>
5
6 char encode_decode(char *c)
7 {
8     char *s = c;
9     int l = strlen(s);
10
11     printf("\nOriginal String: ");
12     puts(s);
13
14     char es[l];
15
16     for (int i = 0; i < l; i++)
17     {
18         if ((int)(s[i]) >= 65 && (int)(s[i]) <= 90)
19         {
20             if ((int)(s[i]) >= 65 && (int)(s[i]) <= 89)
21             {
22                 es[i] = (char)((int)(s[i] + 1));
23             }
24             else if ((int)(s[i]) == 90)
25             {
26                 es[i] = (char)(65);
27             }
28         }
29         else if ((int)(s[i]) >= 97 && (int)(s[i]) <= 122)
30         {

```

C ibrahim_lab9.c ×

C ibrahim_lab9.c > ...

```
31     if ((int)(s[i]) >= 97 && (int)(s[i]) <= 121)
32     {
33         es[i] = (char)((int)(s[i] + 1));
34     }
35     else if ((int)(s[i]) == 122)
36     {
37         es[i] = (char)(97);
38     }
39 }
40 else if ((int)(s[i]) >= 48 && (int)(s[i]) <= 57)
41 {
42     if ((int)(s[i]) >= 48 && (int)(s[i]) <= 56)
43     {
44         es[i] = (char)((int)(s[i] + 1));
45     }
46     else if ((int)(s[i]) == 57)
47     {
48         es[i] = (char)(48);
49     }
50 }
51 else
52 {
53     es[i] = s[i];
54 }
55 }
56
57 // printf ("%d\n", strlen(es));
58 printf("\nEncoded String: ");
59 puts(es);
60
```

C ibrahim_lab9.c ×

C ibrahim_lab9.c > ...

```
61     char ds[1];
62     for (int i = 0; i < 1; i++)
63     {
64         if ((int)(es[i]) >= 65 && (int)(es[i]) <= 90)
65         {
66             if ((int)(es[i]) >= 66 && (int)(es[i]) <= 90)
67             {
68                 ds[i] = (char)((int)(es[i] - 1));
69             }
70             else if ((int)(es[i]) == 65)
71             {
72                 ds[i] = (char)(90);
73             }
74         }
75         else if ((int)(es[i]) >= 97 && (int)(es[i]) <= 122)
76         {
77             if ((int)(es[i]) >= 98 && (int)(es[i]) <= 122)
78             {
79                 ds[i] = (char)((int)(es[i] - 1));
80             }
81             else if ((int)(es[i]) == 97)
82             {
83                 ds[i] = (char)(122);
84             }
85         }
86         else if ((int)(es[i]) >= 48 && (int)(es[i]) <= 57)
87         {
88             if ((int)(es[i]) >= 49 && (int)(es[i]) <= 57)
89             {
90                 ds[i] = (char)((int)(es[i] - 1));
```

```

C ibrahim_lab9.c > ...
91         }
92         else if ((int)(es[i]) == 49)
93         {
94             ds[i] = (char)(57);
95         }
96     }
97     else
98     {
99         ds[i] = es[i];
100     }
101 }
102
103 printf("\nDecoded String: ");
104 puts(ds);
105 }
106
107 int main()
108 {
109     char s[100];
110     printf("Enter the text you want to encode and decode: ");
111     gets(s);
112
113     encode_decode(s);
114
115     return 0;
116 }

```

Output: 9

```

PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab9.c
PS D:\Programming\C Lab Programs HW> .\a.exe
Enter the text you want to encode and decode: my name is ibrahim & my roll is 07

Original String: my name is ibrahim & my roll is 07

Encoded String: nz obnf jt jcsbijn & nz spmm jt 18

Decoded String: my name is ibrahim & my roll is 07
PS D:\Programming\C Lab Programs HW> 

```

Program: 10

```

C ibrahim_lab10.c > main()
1 //complex no. using structure
2 #include <stdio.h>
3 #include <stdlib.h>
4
5 struct Complex
6 {
7     int r;
8     int im;
9 } num1, num2;
10
11 int main()
12 {
13     int t;
14
15     printf("Enter the REAL part of 1st Number: ");
16     scanf("%d", &num1.r);
17     printf("Enter the IMAGINARY part of 1st Number: ");
18     scanf("%d", &num1.im);
19     printf("1st Complex No.: %d + %di\n", num1.r, num1.im);
20
21     printf("Enter the REAL part of 2nd Number: ");
22     scanf("%d", &num2.r);
23     printf("Enter the IMAGINARY part of 2nd Number: ");
24     scanf("%d", &num2.im);
25     printf("2nd Complex No.: %d + %di\n", num2.r, num2.im);
26
27     printf("\nAfter Addition: %d + %di\n", num1.r + num2.r, num1.im + num2.im);
28     printf("\nMultiplication: %d + %di\n", (num1.r * num2.r) - (num1.im * num2.im),
29         (num1.r * num2.im) + (num1.im * num2.r));
30 }

```


Output: 10

```
PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab10.c
PS D:\Programming\C Lab Programs HW> .\a.exe
Enter the REAL part of 1st Number: -9
Enter the IMAGINARY part of 1st Number: 7
1st Complex No.: -9 + 7i
Enter the REAL part of 2nd Number: 5
Enter the IMAGINARY part of 2nd Number: 3
2nd Complex No.: 5 + 3i

After Addition: -4 + 10i

Multiplication: -66 + -15i
PS D:\Programming\C Lab Programs HW> █
```

Program: 11

```
C ibrahim_lab11.c > ...
1  //sum and standard deviation of array
2  #include <stdio.h>
3  float sum(float *a, int n)
4  {
5      float s = 0;
6      for (int i = 0; i < n; i++)
7          s = s + a[i];
8      return s;
9  }
10 float deviation(float t, int n)
11 {
12     return t / n;
13 }
14 int main()
15 {
16     int n;
17     printf("Enter the amount of total numbers you want the sum of: ");
18     scanf("%d", &n);
19     float a[n];
20     for (int i = 0; i < n; i++)
21     {
22         printf("Enter the %dth element: ", i + 1);
23         scanf("%f", &a[i]);
24     }
25     float t = sum(a, n);
26     printf("Sum of Numbers: %.3f\n", t);
27     float x = deviation(t, n);
28     printf("Standard Deviation: %.3f\n", x);
29 }
```

Output: 11

```
PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab11.c
PS D:\Programming\C Lab Programs HW> .\a.exe
Enter the amount of total numbers you want the sum of: 7
Enter the 1th element: 1
Enter the 2th element: 2
Enter the 3th element: 3
Enter the 4th element: 4
Enter the 5th element: 5
Enter the 6th element: 6
Enter the 7th element: 7
Sum of Numbers: 28.000
Standard Deviation: 4.000
PS D:\Programming\C Lab Programs HW> █
```

Program: 12

C ibrahim_lab12.c > ...

```
1  //difference in dates
2
3  #include <stdio.h>
4  #include <string.h>
5  #include <stdlib.h>
6
7  int diffdays (int d1, int d2, int m1, int m2)
8  {
9      int days = 0;
10     int md[12] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
11     if (m1 > m2)
12     {
13         days = days + md[m2 - 1] - d2;
14         for (int i = m2; i < m1 - 1; i++)
15         {
16             days = days + md[i];
17         }
18         days = days + d1;
19     }
20     else if (m2 > m1)
21     {
22         days = days + md[m1 - 1] - d1;
23         for (int i = m1; i < m2 - 1; i++)
24         {
25             days = days + md[i];
26         }
27         days = days + d2;
28     }
29     else if (m1 == m2)
30     {
```

C ibrahim_lab12.c > ...

```
31     days = abs(d1 - d2);
32 }
33 return days;
34 }
35
36 int main()
37 {
38     char date1[20];
39     char date2[20];
40     printf("Enter 1st date in DD-MM-YYYY format : ");
41     scanf("%s", &date1);
42     printf("Enter 2nd date in DD-MM-YYYY format : ");
43     scanf("%s", &date2);
44
45     int i = 0, count = 0;
46     int d1 = 0, d2 = 0, m1 = 0, m2 = 0, y1 = 0, y2 = 0;
47
48     while (date1[i] != '\0' && date2[i] != '\0')
49     {
50         if (date1[i] == '-')
51         {
52             count++;
53             i++;
54         }
55         if (count == 0)
56         {
57             d1 = (d1 * 10) + (date1[i] - '0');
58             d2 = (d2 * 10) + (date2[i] - '0');
59             i++;
60         }
```

```

C ibrahim_lab12.c > ...
61     else if (count == 1)
62     {
63         m1 = (m1 * 10) + (date1[i] - '0');
64         m2 = (m2 * 10) + (date2[i] - '0');
65         i++;
66     }
67     else if (count == 2)
68     {
69         y1 = (y1 * 10) + (date1[i] - '0');
70         y2 = (y2 * 10) + (date2[i] - '0');
71         i++;
72     }
73 }
74
75 //printf("%d & %d & %d\n", d1, m1, y1);
76 //printf("%d & %d & %d\n", d2, m2, y2);
77
78 int ydiff = abs(y1 - y2);
79 int diffd = diffdays(d1, d2, m1, m2);
80
81 int totaldays = ydiff*365 + diffd;
82 printf("\nDifference b/w 2 Dates: %d\n", totaldays);
83
84 return 0;
85 }

```

Output: 12

```

PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab12.c
PS D:\Programming\C Lab Programs HW> .\a.exe
Enter 1st date in DD-MM-YYYY format : 30-11-2022
Enter 2nd date in DD-MM-YYYY format : 30-11-2021

Difference b/w 2 Dates: 365
PS D:\Programming\C Lab Programs HW>

```

Program: 13

```

C ibrahim_lab13.c > ...
1  //file handling program
2
3  #include <stdio.h>
4  #include <stdlib.h>
5  void insert()
6  {
7      FILE *fileptr;
8      char info[1000];
9      fileptr = fopen("student.txt", "a");
10     if (fileptr == NULL)
11     {
12         printf("Unable to Open File!!!\n");
13         exit(EXIT_FAILURE);
14     }
15     printf("Enter Data to Append\n");
16     printf("\n<NAME ROLL_NO. SUB1 SUB2 SUB3 PERCENTAGE> ---> Format\n");
17     fflush(stdin);
18     fgets(info, 1000, stdin);
19     fputs(info, fileptr);
20     printf("Appended Sucessfully\n");
21     fclose(fileptr);
22 }
23 void display()
24 {
25     FILE *fileptr;
26     fileptr = fopen("student.txt", "r");
27     if (fileptr == NULL)
28     {
29         printf("Unable to Open File!!!\n");
30         exit(EXIT_FAILURE);

```

```
31     }
32     char c;
33     printf("\nNAME ROLL_NO. SUB1 SUB2 SUB3 PERCENTAGE\n");
34     while ((c = fgetc(fileptr)) != EOF)
35     {
36         putchar(c);
37     }
38     fclose(fileptr);
39 }
40 void deleter()
41 {
42     FILE *fileptr;
43     fileptr = fopen("student.txt", "r");
44     if (fileptr == NULL)
45     {
46         printf("Unable to Open File!!!\n");
47         exit(EXIT_FAILURE);
48     }
49     FILE *temp;
50     int line;
51     printf("Enter the Line Number that you want to Delete : ");
52     scanf("%d", &line);
53     temp = fopen("temp.txt", "w");
54     if (temp == NULL)
55     {
56         printf("Unable to Open File!!!\n");
57         exit(EXIT_FAILURE);
58     }
59     char c[1000];
60     int count = 1;
61     while (fgets(c, 1000, fileptr) != NULL)
62     {
63         if (line != count)
64         {
65             fputs(c, temp);
66         }
67         count++;
68     }
69     fclose(fileptr);
70     fclose(temp);
71
72     fileptr = fopen("student.txt", "w");
73     if (fileptr == NULL)
74     {
75         printf("Unable to Open File!!!\n");
76         exit(EXIT_FAILURE);
77     }
78     temp = fopen("temp.txt", "r");
79     if (temp == NULL)
80     {
81         printf("Unable to Open File!!!\n");
82         exit(EXIT_FAILURE);
83     }
84     while (fgets(c, 1000, temp) != NULL)
85     {
86         fputs(c, fileptr);
87     }
88     fclose(fileptr);
89     fclose(temp);
90 }
```

```
91 void updater()
92 {
93     FILE *fileptr;
94     fileptr = fopen("student.txt", "r");
95     if (fileptr == NULL)
96     {
97         printf("Unable to Open File!!!\n");
98         exit(EXIT_FAILURE);
99     }
100     FILE *temp;
101     int line;
102     printf("Enter the Line Number that want to Update : ");
103     scanf("%d", &line);
104     char info[1000];
105     printf("Enter Data to Update\n");
106     printf("\n<NAME ROLL_NO. SUB1 SUB2 SUB3 PERCENTAGE> ---> Format\n");
107     fflush(stdin);
108     fgets(info, 1000, stdin);
109     temp = fopen("temp.txt", "w");
110     if (temp == NULL)
111     {
112         printf("Unable to Open File!!!\n");
113         exit(EXIT_FAILURE);
114     }
115     char c[1000];
116     int count = 1;
117     while (fgets(c, 1000, fileptr) != NULL)
118     {
119         if (line != count)
120         {
121             fputs(c, temp);
122         }
123         else
124         {
125             fputs(info, temp);
126         }
127         count++;
128     }
129     fclose(fileptr);
130     fclose(temp);
131     fileptr = fopen("student.txt", "w");
132     if (fileptr == NULL)
133     {
134         printf("Unable to Open File!!!\n");
135         exit(EXIT_FAILURE);
136     }
137     temp = fopen("temp.txt", "r");
138     if (temp == NULL)
139     {
140         printf("Unable to Open File!!!\n");
141         exit(EXIT_FAILURE);
142     }
143     while (fgets(c, 1000, temp) != NULL)
144     {
145         fputs(c, fileptr);
146     }
147     fclose(fileptr);
148     fclose(temp);
149 }
150
```

```

151 int main()
152 {
153     while (1)
154     {
155         printf("\nMENU:\n1.Insert Row\n2.Delete Row\n3.Update Row\n4.Display\n5.Exit\n");
156         int c;
157         scanf("%d", &c);
158         switch (c)
159         {
160             case 1:
161                 insert();
162                 break;
163             case 2:
164                 deleter();
165                 break;
166             case 3:
167                 updater();
168                 break;
169             case 4:
170                 display();
171                 break;
172             case 5:
173                 printf("End of Program\n");
174                 return 0;
175                 break;
176             default:
177                 printf("Invalid Input\n");
178         }
179     }
180 }

```

Output: 13

```

PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab13.c
PS D:\Programming\C Lab Programs HW> .\a.exe

```

MENU:

```

1.Insert Row
2.Delete Row
3.Update Row
4.Display
5.Exit
4

```

NAME ROLL_NO. SUB1 SUB2 SUB3 PERCENTAGE

One 1 50 50 50 50

Two 2 50 50 50 50

MENU:

```

1.Insert Row
2.Delete Row
3.Update Row
4.Display
5.Exit
1

```

Enter Data to Append

<NAME ROLL_NO. SUB1 SUB2 SUB3 PERCENTAGE> ---> Format

ibrahim1 07 10 20 30 40 25

Appended Sucessfully

MENU:

```

1.Insert Row
2.Delete Row
3.Update Row
4.Display
5.Exit
1

```

Enter Data to Append

<NAME ROLL_NO. SUB1 SUB2 SUB3 PERCENTAGE> ---> Format

ibrahim2 17 50 50 50 50

Appended Sucessfully

MENU:

1.Insert Row

2.Delete Row

3.Update Row

4.Display

5.Exit

4

NAME ROLL_NO. SUB1 SUB2 SUB3 PERCENTAGE

One 1 50 50 50 50

Two 2 50 50 50 50

ibrahim1 07 10 20 30 40 25

ibrahim2 17 50 50 50 50

MENU:

1.Insert Row

2.Delete Row

3.Update Row

4.Display

5.Exit

2

Enter the Line Number that you want to Delete : 2

MENU:

1.Insert Row

2.Delete Row

3.Update Row

4.Display

5.Exit

4

NAME ROLL_NO. SUB1 SUB2 SUB3 PERCENTAGE

One 1 50 50 50 50

ibrahim1 07 10 20 30 40 25

ibrahim2 17 50 50 50 50

MENU:

1.Insert Row

2.Delete Row

3.Update Row

4.Display

5.Exit

3

Enter the Line Number that want to Update : 2

Enter Data to Update

<NAME ROLL_NO. SUB1 SUB2 SUB3 PERCENTAGE> ---> Format

ibrahim1 07 100 100 100 100

MENU:

1.Insert Row

2.Delete Row

3.Update Row

4.Display

5.Exit

4

NAME ROLL_NO. SUB1 SUB2 SUB3 PERCENTAGE

One 1 50 50 50 50

ibrahim1 07 100 100 100 100

ibrahim2 17 50 50 50 50

Program: 14

C ibrahim_lab14.c > ...

```
1  //complex numbers - using strings
2
3  #include<stdio.h>
4  #include<stdlib.h>
5  #include<math.h>
6
7  float atof1(char *arr){
8      int i,j,k;
9      float val;
10     char c;
11     i=0;
12     j=0;
13     val=0;
14     k=0;
15     while ((c = *(arr+i))!='\0'){
16         if (c!='.'){
17             val = (val*10)+(c-'0');
18             if (k == 1){
19                 --j;
20             }
21         }
22         if (c=='.'){ if (k == 1) return 0; k=1;}
23         ++i;
24     }
25     val = val*pow(10,j);
26     return val;
27 }
28
29 float RPofCN(char str[])
30 {
31     int i=0,k=0;
32     while(str[i]!='\0')
33     {
34         if(str[i]=='+'||str[i]=='-')
35         {
36             k=1;
37         }
38         i++;
39     }
40     if(k)
41     {
42         float ans1=atof(str);
43         return ans1;
44     }
45     i=0;
46     while(str[i]!='\0')
47     {
48         if(str[i]=='i')
49         {
50             k=1;
51         }
52         i++;
53     }
54     if(k)
55     {
56         float ans2=0.0;
57         return ans2;
58     }
59     else
60     {
```



```

61         float ans3=atof(str);
62         return ans3;
63     }
64 }
65 float IPofCN(char str[])
66 {
67     int i=0,si=0,k=0;
68     while(str[i]!='\0')
69     {
70         if(str[i]=='+'||str[i]=='-')
71         {
72             k=1;
73             if(str[i]=='-')
74             {
75                 si=-1;
76             }
77             break;
78         }
79         i++;
80     }
81     i++;
82     int j=0;
83     char st[10];
84     if(k)
85     {
86         while(str[i]!='\0')
87         {
88             st[j++]=str[i++];
89         }
90         float ans1=atof(st);
91         if(si==-1)
92         {
93             ans1=0.0-ans1;
94         }
95         return ans1;
96     }
97     i=0,si=0;
98     while(str[i]!='\0')
99     {
100         if(str[i]=='i')
101         {
102             k=1;
103         }
104         i++;
105     }
106     if(k)
107     {
108         float ans2=atof(str);
109         return ans2;
110     }
111     else
112     {
113         float ans3=0.0;
114         return ans3;
115     }
116 }
117 float addTwoNo(float n,float m)
118 {
119     float ans=n+m;
120     return ans;

```

```

121 }
122 float subTwoNo(float n,float m)
123 {
124     float ans=n-m;
125     return ans;
126 }
127 void add(char a[],char b[])
128 {
129     float RPofAns=RPofCN(a)+RPofCN(b);
130     float IPofAns=IPofCN(a)+IPofCN(b);
131     if(IPofAns<0)
132     {
133         printf("sum is ---> %.2f %.2f%c\n",RPofAns,IPofAns,'i');
134     }
135     else
136     {
137         printf("sum is ---> %.2f %c %.2f%c\n",RPofAns,'+',IPofAns,'i');
138     }
139 }
140 void subtract(char a[],char b[])
141 {
142     float RPofAns=RPofCN(a)-RPofCN(b);
143     float IPofAns=IPofCN(a)-IPofCN(b);
144     if(IPofAns<0)
145     {
146         printf("Difference is ---> %.2f %.2f%c\n",RPofAns,IPofAns,'i');
147     }
148     else
149     {
150         printf("Difference is ---> %.2f %c %.2f%c\n",RPofAns,'+',IPofAns,'i');
151     }
152 }
153
154 int main()
155 {
156     char str1[20];
157     printf("Enter 1st complex no: ");
158     gets(str1);
159     printf("\n");
160     char str2[20];
161     printf("Enter 2nd complex no: ");
162     gets(str2);
163     printf("\n");
164     add(str1,str2);
165     subtract(str1,str2);
166 }

```

Output: 14

```
PS D:\Programming\C Lab Programs HW> gcc ibrahim_lab14.c
PS D:\Programming\C Lab Programs HW> .\a.exe
Enter 1st complex no: 20 - 10i

Enter 2nd complex no: 10 + 10i

sum is ---> 30.00 + 0.00i
Difference is ---> 10.00 -20.00i
PS D:\Programming\C Lab Programs HW> █
```
