Course Code	Course Title
CS448	OPERATING SYSTEM

Student Details: Student Name: Asha sahu
Roll/Reg No: 20252651013
Email: 20252651013@iiitvadodara.ac.in
Mobile: 9340199074

Question 1	**Write a program to display the address of an integer variable, character variable and string variable using pointers.**
Program	

```c
#include <stdio.h>
int main()
{
    int a;
    int *p;
    char ch;
    char *p1;
    char str[20];
    char *p2;

    printf("Enter any value of a: ");
    scanf("%d", &a);
    printf("Enter any character: ");
    scanf("%c \n", &ch);
    printf("Enter any string value: ");
    scanf("%s \n", str);

    p = &a;
    p1 = &ch;
    p2 = str;

    printf("Address of a using pointer p: %p\n", (void *)p);
    printf("Address of ch using pointer p1: %p\n", (void *)p1);
    printf("Address of s using pointer p2: %p\n", (void *)p2);

    return 0;
}
```

Output:

```
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q1.c -o q1.exe
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q1.exe
```

	Enter any value of a: 45 Enter any character: f Enter any string value: komal Address of a using pointer p: 0061FF10 Address of ch using pointer p1: 0061FF0F Address of s using pointer p2: 0061FEFB

Question 2	**Write a program to add two integers using pointers.**

Program	`#include <stdio.h>`
	`int main()`
	`{`
	`  int a, b;`
	`  int *ptr1;`
	`  int *ptr2;`
	
	`  printf("Enter values of a and b: ");`
	`  scanf("%d %d", &a, &b);`
	
	`  ptr1 = &a;`
	`  ptr2 = &b;`
	
	`  printf("sum of %d+%d = %d \n ", *ptr1, *ptr2, *ptr1 + *ptr2);`
	
	`  return 0;`
	`}`

Output:	PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q2.c -o q2.exe PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q2.exe Enter values of a and b: 34 67 sum of 34+67 = 101

Question 3	**Write a program to find the maximum number between two numbers using pointers.**

Program	`#include <stdio.h>`
	`int main()`
	`{`
	`  int a, b;`
	`  int *ptr1;`
	`  int *ptr2;`
	
	`  printf("Enter values of a and b: ");`
	`  scanf("%d %d", &a, &b);`

```
    ptr1 = &a;
    ptr2 = &b;

    if (*ptr1 > *ptr2)
        printf("%d is greater", *ptr1);
    else
        printf("%d is greater", *ptr2);

    return 0;
}
```

Output:

```
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q3.c -o q3.exe
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q3.exe
Enter values of a and b: 87 4
87 is greater
```

Question 4	**Write a program to store n elements in an array and print the elements using pointers.**

Program

```c
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int n; // size of array
    int *arr;

    printf("Enter no of elements: ");
    scanf("%d", &n);

    arr = (int *)malloc(n * sizeof(int));

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

    for (int i = 0; i < n; i++)
    {
        printf("arr[%d] = %d \n", i, *(arr + i));
    }
```

```
    free(arr);
    return 0;
}
```

Output:

```
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q4.c -o q4.exe
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q4.exe
Enter no of elements: 3
Enter value for arr[0] : 7
Enter value for arr[1] : 34
Enter value for arr[2] : 2
arr[0] = 7
arr[1] = 34
arr[2] = 2
```

Question 5	**Write a program to calculate the length of the string using pointer.**

Program

```c
#include <stdio.h>
int main()
{
    char s[100];
    char *ptr;
    int i = 0;

    printf("Enter any string: ");
    scanf("%s", s);

    int len = 0;

    ptr = s;
    for (i = 0; *(ptr + i) != '\0'; i++)
    {
        len++;
    }
    printf("Length is %d", len);

    return 0;
}
```

Output:

```
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q5.c -o q5.exe
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q5.exe
Enter any string: ghdufjhdjgd
Length is 11
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1>
```

Question 6	Write a program to sort an array using pointers.
Program	

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

    int arr[n];
    int *ptr = arr;
    int temp;

    printf("Array elements: ");
    for (int i = 0; i < n; i++)
        scanf("%d", (ptr + i));

    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n - i; j++)
        {
            if (*(ptr + j) > *(ptr + j + 1))
            {
                temp = *(ptr + j);
                *(ptr + j) = *(ptr + j + 1);
                *(ptr + j + 1) = temp;
            }
        }
    }

    printf("Sorted array is : ");
    for (int i = 0; i < n; i++)
    {
        printf("%d \t", *(ptr + i));
    }

    return 0;
}
```

Output:

```
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q6.c -o q6.exe
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q6.exe
```

	Enter the size of Array: 5 Array elements: 1 9 5 2 9 Sorted array is : 1 2 5 9 9

Question 7	**Write a program in C to create and store information in a text file.**

Program	`#include <stdio.h>`
	
	`int main()`
	`{`
	`  FILE *fp;`
	`  char words[50];`
	
	`  fp = fopen("file.txt", "w");`
	`  if (fp == NULL)`
	`  {`
	`  printf("Invalid!");`
	`  return 1;`
	`  }`
	
	`  printf("Write something: ");`
	`  gets(words);`
	`  fprintf(fp, "%s", words);`
	`  fclose(fp);`
	
	`  printf("Information added successfully.\n");`
	
	`  return 0;`
	`}`

Output:	PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q7.c -o q7.exe PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q7.exe Write something: hey this is assignment of operating system's first lab. Information added successfully.

Question 8	**Write a program in C to count a number of words and characters in a file.**

Program	`#include <stdio.h>`
	`#include <ctype.h>`
	
	`int main()`
	`{`

```c
    FILE *fp;
    char ch;
    int characters = 0, words = 0;
    int Word = 0;

    fp = fopen("file.txt", "r");

    if (fp == NULL)
    {
        printf("Error! File not found.\n");
        return 1;
    }

    while ((ch = fgetc(fp)) != EOF)
    {
        characters++;

        if (isalpha(ch))
        {
            Word = 0;
        }
        else if (Word == 0)
        {
            Word = 1;
            words++;
        }
    }
    fclose(fp);

    printf("Characters: %d\n", characters);
    printf("Words: %d\n", words);

    return 0;
}
```

Output:

```
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q8.c -o q8.exe
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q8.exe
Characters: 41
Words: 8
```

Question 9	**Write a program in C to merge two files and write it in a new file.**
Program	`#include <stdio.h>`

```c
int main()
{
    FILE *f1, *f2, *merge;
    char ch;

    f1 = fopen("f1.txt", "r");
    if (f1 == NULL)
    {
        printf("Cannot open f1.txt \n");
        return 1;
    }

    f2 = fopen("f2.txt", "r");
    if (f2 == NULL)
    {
        printf("Cannot open f2.txt \n");
        fclose(f1);
        return 1;
    }

    merge = fopen("merge.txt", "w");
    if (merge == NULL)
    {
        printf("Cannot create merge.txt \n");
        fclose(f1);
        fclose(f2);
        return 1;
    }

    while ((ch = fgetc(f1)) != EOF)
    {
        fputc(ch, merge);
    }

    while ((ch = fgetc(f2)) != EOF)
    {
        fputc(ch, merge);
    }

    printf("Files merged succesfully into merge.txt \n");
```

```
    fclose(f1);
    fclose(f2);
    fclose(merge);

    return 0;
}
```

Output:

```
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q9.c -o q9.exe
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q9.exe
Files merged succesfully into merge.txt
```

Question 10

Write a C program that prints out all its command-line arguments except those that begin with a dash. Hint: find out about argc and argv. If all arguments begin with a dash, or if there are no arguments at all, print nothing.

Program

```
#include <stdio.h>
int main(int argc, char *argv[])
{
    int i;

    printf("argc = %d\n", argc);

    for (i + 1; i < argc; i++)
    {
        printf("argv[%d] = %s\n", i, argv[i]);
    }
    return 0;
}
```

Output:

```
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q10.c -o q10.exe
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q10.exe asha sahu
argc = 3
```

Question 11

Write C program which will read characters containing non-alphanumericcharacters from its standard input stdin, count the number of NON-alphabetic, lower case, and upper case characters including the newlines that is, count those characters not in the a-z range nor in the A–Z range, and display the count onto the screen. Remove all the non-alphanumeric characters from the string and write out all characters into the file. Take the input from user and write the output into a file.

Program

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

int main()
```

```c
{
    FILE *fp;
    char ch;
    int lowC = 0, upC = 0, other = 0;

    fp = fopen("file.txt", "a");
    if (fp == NULL)
    {
        printf("File could not be opened.\n");
        return 1;
    }

    printf("Enter text (Ctrl+Z then Enter to stop on Windows):\n");

    while ((ch = getchar()) != EOF)
    {
        if (isalpha(ch))
        {
            if (islower(ch))
                lowC++;
            else
                upC++;
        }
        else
        {
            other++;
        }

        if (isalnum(ch))
        {
            fputc(ch, fp);
        }
    }

    fclose(fp);

    printf("\nCharacter Count Summary:\n");
    printf("Lowercase characters : %d\n", lowC);
    printf("Uppercase characters : %d\n", upC);
    printf("Non-alphabetic chars : %d\n", other);
```

```
    printf("\nData written to existing file: file.txt\n");

    return 0;
}
```

Output:

```
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q11.c -o q11.exe
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q11.exe
Enter text (Ctrl+Z then Enter to stop on Windows):
this is prog 11 of lab assignment 01
```

Question 12	**Write a C program to read the input from the file generated from previous question and rotate all the alphabets by 13 positions.**

Program

```c
#include <stdio.h>
#include <ctype.h>

char rot_13(char ch)
{
    if (ch >= 'a' && ch <= 'z')
        return 'a' + (ch - 'a' + 13) % 26;
    else if (ch >= 'A' && ch <= 'Z')
        return 'A' + (ch - 'A' + 13) % 26;
    else
        return ch;
}

int main()
{
    FILE *f_in, *f_out;
    int ch;

    f_in = fopen("file.txt", "r");
    f_out = fopen("rot_13.txt", "w");

    if (f_in == NULL || f_out == NULL)
    {
        printf("Cannot open file!\n ");
        return 1;
    }

    while ((ch = fgetc(f_in)) != EOF)
    {
```

```
        fputc(rot_13((char)ch), f_out);
    }

    fclose(f_in);
    fclose(f_out);

    printf("rot_13 conversion completed.\n ");
    printf("Check file rot_13.txt for result.\n ");

    return 0;
}
```

Output:

```
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> gcc q12.c -o q12.exe
PS C:\Users\ashas\OneDrive\Desktop\sem2\OS\OS lab\Lab1> .\q12.exe
rot_13 conversion completed.
 Check file rot_13.txt for result.
```