**Practical-3**

**1. Write a program in C to implement arrays of pointers and pointers to arrays.**

Aim:

To differentiate between the concept of array of pointers and pointer to array .

Theory:

In this practical we took an array of pointers and stored the address of the values entered by the user also we displayed them in the output using a single pointer to access other elements of the array.

Program/Code:

#include <stdio.h>

int main()

{

    // array of pointers

    int \*p[5];

    int arr[5];

    printf("Enter 5 integers where the pointer should be pointing=\n");

    for (int i=0;i<5;i++)

    {

        scanf("%d",&arr[i]);

    }

    for(int i=0;i<5;i++)

    {

        p[i]=&arr[i];

    }

    printf("The array of integers contains following elements=\n");

    for (int i=0;i<5;i++)

    {

        printf("%d\n",\*p[i]);

    }

    // pointer to array

    int \*q=arr;

    printf("Printing the array with the use of pointer=");

    for (int i=0;i<5;i++)

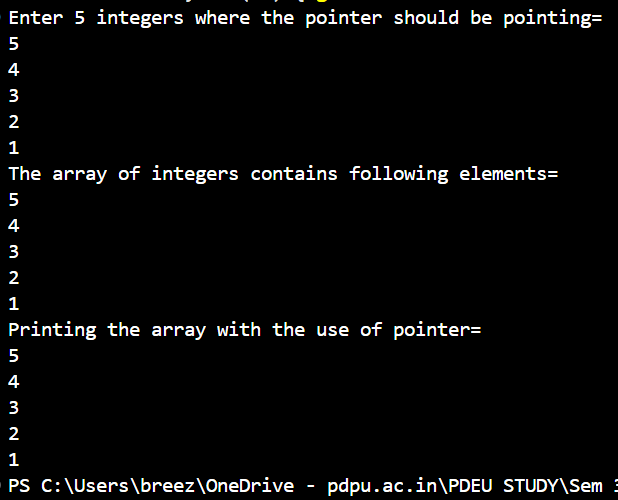
    {

        printf("\n%d",\*(q+i));

    }

}

Output:



Time Complexity:

O(1)

**2.** **Write a program in C to implement pointers to structures.**

Aim:

To showcase the implementation of the pointers in structure.

Theory:

In this practical we used pointer to point to the values stored in the address of struct by passing the address to the function and storing it to a pointer.

Program/Code:

#include <stdio.h>

struct student

{

    char student\_name[100];

    int student\_rollno;

    char student\_address[100];

};

void print\_info(struct student \*s1)

{

    printf("Name of student is= %s \n",(\*s1).student\_name);

    printf("Roll no of student = %d \n",(\*s1).student\_rollno);

    printf("The address of student is = %s \n",(\*s1).student\_address);

}

void main()

{

    struct student s;

    printf("Enter the name of the student=\n");

    scanf("%s",s.student\_name);

    printf("Enter the roll no of student=\n");

    scanf("%d",&s.student\_rollno);

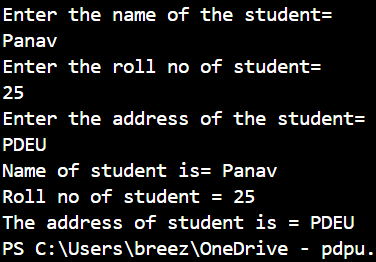
    printf("Enter the address of the student=\n");

    scanf("%s",s.student\_address);

    print\_info(&s);

}

Output:



Time Complexity:

O(n)

**3.** **Write a program in C to perform swapping of two numbers by passing addresses of the variables to the functions.**

Aim:

To swap the integer using pointers.

Theory:

In this practical we swapped numbers using the address of it .

Program/Code:

#include<stdio.h>

int swap(int \*a,int \*b)

{

  int temp=\*a;

  \*a=\*b;

  \*b=temp;

  printf("The swapped number is= %d %d",\*a,\*b);

  return 0;

}

int main()

{

  int p,q;

  printf("Enter two number=");

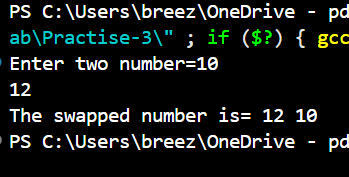
  scanf("%d %d",&p,&q);

  swap(&p,&q);

  return 0;

}

Output:



Time Complexity:

O(n)