| **Name** | Jhaveri Varun Nimitt |
| --- | --- |
| **UID no.** | 2023800042 |
| **Experiment No.** | 9 |

| **AIM:** | **Demonstrate the use of pointers to solve a given problem.** |
| --- | --- |
| **Program 1** | |
| **PROBLEM STATEMENT :** | Write a program to reverse the position of all elements in the array using  pointers. |
| **PROGRAM:** | #include <stdio.h>  */\*void reverse(int \*arr, int size)  {  int \*start = arr;  int \*end = arr + size - 1;   while (start < end)   {  int temp = \*start;  \*start = \*end;  \*end = temp;  start++;  end--;  }   }\*/*  void reverse (int \*arr, int size) {  int temp;  for (int i = 0; i < size/2; i++)  {  temp = arr[i];  arr[i] = arr[size - i - 1];  arr[size - i - 1] = temp;  } }  void print(int \*arr, int size)  {  for (int i = 0; i < size; i++)  {  printf("%d ", arr[i]);  } }  int main()  {  int arr[] = {10,20,30,40,50,60,70,80,90,100};  int size = sizeof(arr) / sizeof(arr[0]);   printf("Array before reversing: ");  print(arr, size);   reverse(arr, size);   printf("\nArray after reversing: ");  print(arr, size);    return 0; } |
| **RESULT:** | |
| **Program 2** | |
| **PROBLEM STATEMENT :** | Write a program to perform matrix addition using pointers. |
| **PROGRAM:** | #include <stdio.h>  void add(int mat1[][3], int mat2[][3], int result[][3], int rows, int columns) {  for (int i = 0; i < rows; i++)  {  for (int j = 0; j < columns; j++)  {  \*(\*(result+i)+j) = \*(\*(mat1+i)+j) + \*(\*(mat2+i)+j);  }  } }  void printMatrix(int matrix[][3], int rows, int columns) {  for (int i = 0; i < rows; i++)  {  for (int j = 0; j < columns; j++)  {  printf("%d ", matrix[i][j]);  }  printf("\n");  } }  int main() {  int matrix1[3][3] = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};  int matrix2[3][3] = {{9, 8, 7}, {6, 5, 4}, {3, 2, 1}};  int result[3][3];  *//scanf ("%d", \*(n+i)+g);")*  add(matrix1, matrix2, result, 3, 3);    printf("Matrix 1:\n");  printMatrix(matrix1, 3, 3);    printf("\nMatrix 2:\n");  printMatrix(matrix2, 3, 3);    printf("\nResult:\n");  printMatrix(result, 3, 3);    return 0; } |
| **RESULT:** | |
| **CONCLUSION:** | **I have understood what double pointers are, how to use them, and how to solve array based questions with pointers.** |