|  |  |
| --- | --- |
| **Practical: 1** | **Write a program that implement array operations a) Insertion b) Deletion** |
| **25-07-2024** |

**1a) Inserting the element into the array at any specific position.**

**Program**:

*#include* <iostream>

using namespace std;

int main()

{

    int arreaySize\_input;

    cout << "Enter the size of array: " << endl;

    cin >> arreaySize\_input;

    int arraySize = arreaySize\_input;

    int array[arraySize + 1];

    int i;

    cout << "Enter the array elements: " << endl;

*for* (i = 0; i < arraySize; i++)

    {

        cin >> array[i];

    }

    int position, element;

    cout << "Enter the position to insert the new element (0 to " << arraySize << "): ";

    cin >> position;

    cout << "Enter the element to insert: ";

    cin >> element;

*for* (i = arraySize; i > position; i--)

    {

        array[i] = array[i - 1];

    }

    array[position] = element;

    arraySize++;

    cout << "The array elements after insertion are: ";

*for* (i = 0; i < arraySize; i++)

    {

        cout << array[i] << " ";

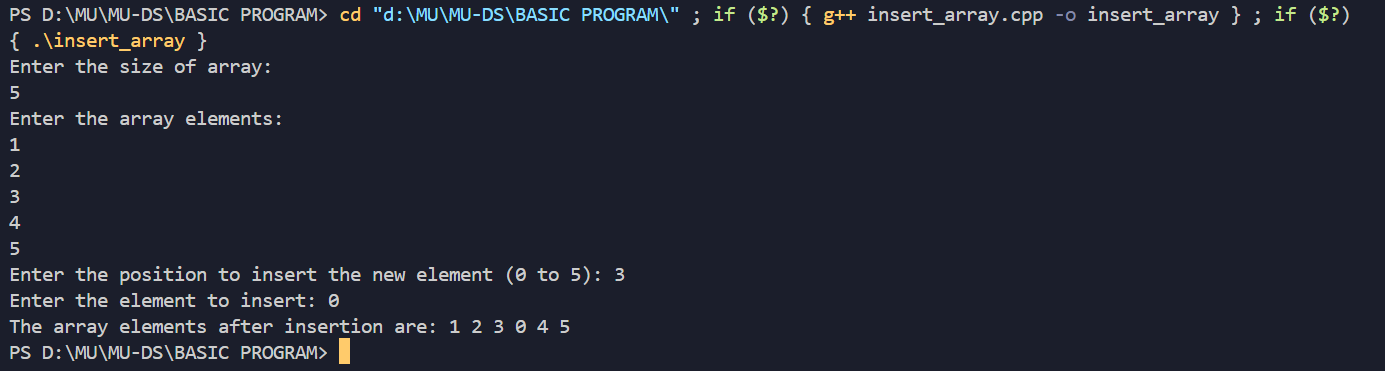
    }

    cout << endl;

*return* 0;

}

**Output:**



**1b) Deleting the element from the array.**

**Program:**

*#include* <iostream>

using namespace std;

int main()

{

    int arreaySize\_input;

    cout << "Enter the size of array: " << endl;

    cin >> arreaySize\_input;

    int arraySize = arreaySize\_input;

    int array[arraySize];

    int i;

    cout << "Enter the array elements: " << endl;

*for* (i = 0; i < arraySize; i++)

    {

*// Read elements into the array*

        cin >> array[i];

    }

    int position;

    cout << "Enter the position of the element to delete (0 to " << arraySize - 1 << "): ";

    cin >> position;

*if* (position < 0 || position >= arraySize)

    {

        cout << "Invalid position!" << endl;

    }

*else*

    {

*// Shift left*

*for* (i = position; i < arraySize - 1; i++)

        {

            array[i] = array[i + 1];

        }

        arraySize--;

*// Update array*

        cout << "The array elements after deletion are: ";

*for* (i = 0; i < arraySize; i++)

        {

            cout << array[i] << " ";

        }

        cout << endl;

    }

*return* 0;

}

**Output:**

****

**Conclusion:**

Deleting an element from an array involves shifting subsequent elements to fill the gap, maintaining contiguity, and has a time complexity of O(n). Inserting an element at a specific position requires shifting elements to the right to make room, also with a time complexity of O(n), and may require resizing the array if it is full.

|  |  |
| --- | --- |
| **Practical: 2** | **Write a program that implements the following sorting**  **a) Bubble sort b) Insertion sort c) Selection sort** |
| **01-08-2024** |

**2 a) Program to implement Bubble sort.**

**Program:**

*#include* <iostream>

using namespace std;

int main()

{

    int i, arr[50], n, x, y;

    cout << "Enter the size of array:";

    cin >> n;

    cout << "Enter the elements in an array:";

*for* (i = 0; i < n; i++)

    {

        cin >> arr[i];

    }

*for* (i = 0; i < n; i++)

    {

*for* (int j = 0; j < n; j++)

        {

*if* (arr[j] > arr[j + 1])

            {

                int temp = arr[j];

                arr[j] = arr[j + 1];

                arr[j + 1] = temp;

            }

        }

    }

*for* (i = 0; i < n; i++)

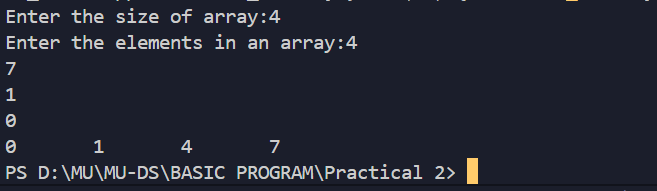
    {

        cout << arr[i] << "\t";

    }

}

**Output:**



**2 b) Program to implement Insertion sort.**

**Program:**

*#include* <iostream>

using namespace std;

int main()

{

    int i, arr[50], n, j, current;

    cout << "Enter the size of array:";

    cin >> n;

    cout << "Enter the elements in an array:";

*for* (i = 0; i < n; i++)

    {

        cin >> arr[i];

    }

*for* (i = 1; i < n; i++)

    {

        current = arr[i];

        j = i - 1;

*while* (arr[j] > current && j >= 0)

        {

            arr[j + 1] = arr[j];

            j--;

        }

        arr[j + 1] = current;

    }

*for* (i = 0; i < n; i++)

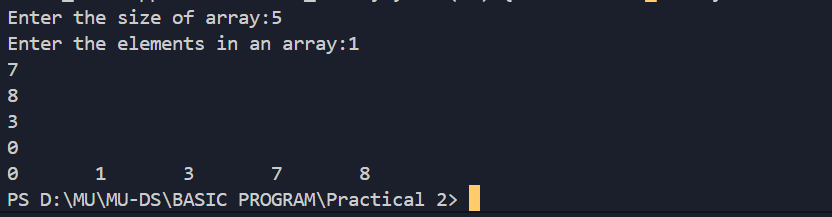
    {

        cout << arr[i] << "\t";

    }

}

**Output:**

****

**2 c) Program to implement Selection sort.**

**Program:**

*#include* <iostream>

using namespace std;

int main()

{

    int i, arr[50], n, x, y;

    cout << "Enter the size of array:";

    cin >> n;

    cout << "Enter the elements in an array:";

*for* (i = 0; i < n; i++)

    {

        cin >> arr[i];

    }

*for* (i = 0; i < n - 1; i++)

    {

*for* (int j = i + 1; j < n; j++)

        {

*if* (arr[i] > arr[j])

            {

                int temp = arr[i];

                arr[i] = arr[j];

                arr[j] = temp;

            }

        }

    }

*for* (i = 0; i < n; i++)

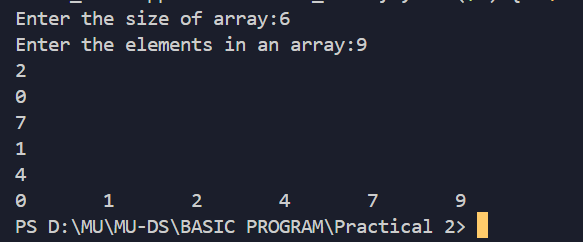
    {

        cout << arr[i] << "\t";

    }

}

**Output:**

****

**Conclusion:**

In this experiment we have learned how to Sort elements in an Array using Bubble Sorting,Insertion Sorting and Selection sorting.