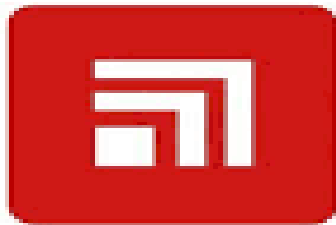


# **Lab Programs**

## **Algorithm Design and Implementation**

**Riya**

**2210997197**



**CHITKARA**  
**UNIVERSITY**  

---

**PUNJAB**

**1. Write down a menu driven program to insert element in an array.**

**(i) Insertion of element at beginning of an array.**

```
#include<iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    cout << "riya" << endl;
```

```
    cout << "2210997197" << endl;
```

```
    int arr[10], size, i, newelement;
```

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

```
    cin >> size;
```

```
    if (size > 10) {
```

```
        cout << "Array size exceeds the maximum allowed size of 10." << endl;
```

```
        return 1;
```

```
    }
```

```
    cout << "Enter the Elements of the array: ";
```

```
    for (i = 0; i < size; i++) {
```

```
        cin >> arr[i];
```

```
    }
```

```
    cout << "Enter the Element that you want to insert at beginning: ";
```

```
    cin >> newelement;
```

```

if (size >= 10) {

    cout << "Cannot insert new element, array is at full capacity." << endl;

    return 1;

}

for (i = size; i > 0; i--) {

    arr[i] = arr[i - 1];

}

arr[0] = newelement;

size++;

cout << "After inserting the new Element at the beginning, Array Elements are:" << endl;

for (i = 0; i < size; i++) {

    cout << arr[i] << endl;

}

Return 0;

}

```

| Run | Output  |
|-----|---|
|     | <pre> /tmp/aumYCaZWkz.o riya 2210997197 Enter the size of the array: 5 Enter the Elements of the array: 1 2 3 4 5 Enter the Element that you want to insert at beginning: 12 After inserting the new Element at the beginning, Array Elements are: 12 1 2 3 4 5  === Code Execution Successful === </pre> |

**(ii) Insertion of element at end of an array.**

```
#include<iostream>

using namespace std;

int main() {
    cout << "riya" << endl;
    cout << "2210997197" << endl;

    int arr[5], n, newelement, i;

    cout << "Enter the size of the array (up to 5): ";
    cin >> n;

    if (n > 5) {
        cout << "Array size exceeds the maximum allowed size of 5." << endl;
        return 1;
    }

    cout << "Enter the Elements of the array: ";
    for(i = 0; i < n; i++) {
        cin >> arr[i];
    }

    cout << "Enter the Element that you want to insert at the end of the array: ";
    cin >> newelement;

    // Insert the new element at the end of the array
    if (n < 5) { // Ensure there is space to insert
        arr[n] = newelement;
        n++;
    } else {
```

```

        cout << "Array is full. Cannot insert new element." << endl;

        return 1;
    }

    cout << "After inserting at the end, Array Elements are:" << endl;
    for(i = 0; i < n; i++) {
        cout << arr[i] << endl;
    }

    return 0;
}

```

| Run | Output   |
|-----|--|
|     | <pre> /tmp/fm3hFGhmB3.o riya 2210997197 Enter the size of the array (up to 5): 5 Enter the Elements of the array: 1 2 3 4 5 Enter the Element that you want to insert at the end of the array: 6 After inserting at the end, Array Elements are: 1 2 3 4 5 6  === Code Execution Successful === </pre> |

**(iii) Insertion of element at a specified location within an array.**

```

#include <iostream>
using namespace std;

int main() {
    cout << "riya" << endl;
    cout << "2210997197" << endl;

    int arr[10], n, i, newelement, x;

```

```

cout << "Enter the size of the array (up to 10): ";
cin >> n;

if (n < 0 || n > 10) {
    cout << "Invalid array size. It must be between 0 and 10." << endl;
    return 1;
}

cout << "Enter the Elements of the Array: ";
for (i = 0; i < n; i++) {
    cin >> arr[i];
}

cout << "Enter the New Element: ";
cin >> newelement;

cout << "Enter the specific location to insert the new Element (0 to " << n << "): ";
cin >> x;

if (x < 0 || x > n) {
    cout << "Invalid location. It must be between 0 and " << n << "." << endl;
    return 1;
}

if (n >= 10) {
    cout << "Array is full. Cannot insert new element." << endl;
    return 1;
}

for (i = n; i > x; i--) {
    arr[i] = arr[i - 1];
}

arr[x] = newelement;
n++;

cout << "After inserting at the specific location, Array Elements are:" << endl;
for (i = 0; i < n; i++) {
    cout << arr[i] << endl;
}

return 0;
}

```

```
/tmp/gankTViHpr.o
riya
2210997197
Enter the size of the array (up to 10): 5
Enter the Elements of the Array: 1 2 3 4 5
Enter the New Element: 8
Enter the specific location to insert the new Element (0 to 5): 3
After inserting at the specific location, Array Elements are:
1
2
3
8
4
5

=== Code Execution Successful ===
```

## 2. Write down a menu driven program to perform delete operation in an array

### (i) Deletion of element at beginning of an array.

```
#include<iostream>

using namespace std;

int main() {

    cout << "riya" << endl;

    cout << "2210997197" << endl;

    int arr[5], n, i;

    cout << "Enter the Size of the Array (up to 5): ";

    cin >> n;

    if (n < 0 || n > 5) {

        cout << "Invalid size. The size must be between 0 and 5." << endl;
```

```
        return 1;
    }

    if (n == 0) {
        cout << "The array is empty. Nothing to delete." << endl;
        return 1;
    }

    cout << "Enter the Elements of the Array: ";
    for (i = 0; i < n; i++) {
        cin >> arr[i];
    }

    for (i = 0; i < n - 1; i++) {
        arr[i] = arr[i + 1];
    }

    n--; // Decrease size after deletion

    cout << "After Deletion, Array Elements are:" << endl;
    for (i = 0; i < n; i++) {
        cout << arr[i] << endl;
    }

    return 0;
```



}

| Run | Output  |
|-----|---|
|     | <pre>/tmp/dlnTZr2Im8.o riya 2210997197 Enter the Size of the Array (up to 5): 5 Enter the Elements of the Array: 1 2 3 4 5 After Deletion, Array Elements are: 2 3 4 5  === Code Execution Successful ===</pre> |

**(ii) Deletion of element at end of an array.**

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    cout << "riya" << endl;
```

```
    cout << "2210997197" << endl;
```

```
    int arr[5], i, size;
```

```
    cout << "Enter the size of the array (up to 5): ";
```

```
    cin >> size;
```

```
    if (size < 1 || size > 5) {
```

```

        cout << "Invalid size. The size must be between 1 and 5." << endl;
        return 1;
    }

    cout << "Enter the Elements of the Array: ";
    for (i = 0; i < size; i++) {
        cin >> arr[i];
    }
    size--;

    cout << "After Deleting the Last Element, the Array is:" << endl;
    for (i = 0; i < size; ++i) {
        cout << arr[i] << " ";
    }
    cout << endl;

    return 0;
}

```

| Run | Output  |
|-----|---|
|     | <pre> /tmp/cN8Cj8Lq03.o riya 2210997197 Enter the size of the array (up to 5): 5 Enter the Elements of the Array: 1 2 3 4 5 After Deleting the Last Element, the Array is: 1 2 3 4 </pre> |
|     | <pre> === Code Execution Successful === </pre>  |

**(iii) Deletion of element at a specified location within an array.**

```
#include <iostream>
using namespace std;

int main() {
    cout << "riya" << endl;
    cout << "2210997197" << endl;

    int arr[5], size, i;
    int delete_element;

    cout << "Enter the Size of the Array (up to 5): ";
    cin >> size;

    if (size < 1 || size > 5) {
        cout << "Invalid size. The size must be between 1 and 5." << endl;
        return 1;
    }

    cout << "Enter the Elements of the Array: ";
    for (i = 0; i < size; i++) {
        cin >> arr[i];
    }

    cout << "Enter the index of the Element that you want to Delete (0 to " << size - 1 << "): ";
    cin >> delete_element;

    if (delete_element < 0 || delete_element >= size) {
        cout << "Invalid index. It must be between 0 and " << size - 1 << "." << endl;
        return 1;
    }

    for (i = delete_element; i < size - 1; ++i) {
        arr[i] = arr[i + 1];
    }

    size--; // Decrease size after deletion

    cout << "After Deleting from the specific location, Elements of the Array are:" << endl;
    for (i = 0; i < size; ++i) {
        cout << arr[i] << " ";
    }
    cout << endl;

    return 0;
}
```

| Run | Output   |
|-----|--|
|     | <pre>/tmp/UYbUnGm5Tb.o riya 2210997197 Enter the Size of the Array (up to 5): 5 Enter the Elements of the Array: 1 2 3 4 5 Enter the index of the Element that you want to Delete (0 to 4): 4 After Deleting from the specific location, Elements of the Array are: 1 2 3 4  === Code Execution Successful ===</pre> |

### 3. Write down a menu driven program to perform Search operation in an array

- (i) **Generate a program to search and display all locations of a given value/element using Linear search.**

```
#include <iostream>
using namespace std;
```

```
void search(int arr[], int n, int x) {
    bool found = false;
    for (int i = 0; i < n; i++) {
        if (arr[i] == x) {
            cout<<"name:"<<"riya"<<endl;
            cout<<"roll no:"<<2210997197<<endl;
            cout << "Element " << x << " found at index " << i << endl;
            found = true;
        }
    }
    if (!found) {
        cout << "Element " << x << " not found in the array." << endl;
    }
}
```

```
int main() {
    int arr[] = {2, 3, 4, 10, 40};
    int x = 10;
    int n = sizeof(arr) / sizeof(arr[0]);

    search(arr, n, x);

    return 0;
}
```

| Output   |
|--|
| <pre>/tmp/ns4Zmn0W1H.o name:riya roll no:2210997197 Element 10 found at index 3  === Code Execution Successful ===</pre> |

(ii). create a program to check for the presence of an element in a given array using Binary Search.

```
#include <iostream>

using namespace std;

int binarySearch(int arr[], int low, int high, int x) {
    while (low <= high) {
        int mid = low + (high - low) / 2;

        if (arr[mid] == x)
            return mid; // Element found at index 'mid'
        else if (arr[mid] < x)
            low = mid + 1; // Search right half
        else
```

```
        high = mid - 1; // Search left half
    }

    return -1; // Element not found
}

int main() {

    cout << "riya" << endl;

    cout << "2210997197" << endl;

    int arr[] = {2, 3, 4, 10, 40};

    int x = 10;

    int n = sizeof(arr) / sizeof(arr[0]);

    int result = binarySearch(arr, 0, n - 1, x);

    if (result != -1)

        cout << "Element is present at index " << result << endl;

    else

        cout << "Element is not present in the array." << endl;

    return 0;
```

}

| Run | Output  |
|-----|---|
|     | <pre>^ /tmp/YBjlmCwW6i.o riya 2210997197 Element is present at index 3  === Code Execution Successful ===</pre> |

**4. Write down a menu driven program to perform Sort operation in an array**

**(i). Write a code to arrange the elements of array in ascending/descending order using Bubble Sort algorithm.**

```
#include <iostream>
using namespace std;

void bubbleSortAscending(int arr[], int n) {
    for (int i = 0; i < n - 1; i++) {
        for (int j = 0; j < n - i - 1; j++) {
```

```
        if (arr[j] > arr[j + 1]) {  
            swap(arr[j], arr[j + 1]);  
        }  
    }  
}  
}
```

```
void printArray(const int arr[], int size) {  
    for (int i = 0; i < size; i++) {  
        cout << arr[i] << " ";  
    }  
    cout << endl;  
}
```

```
int main() {  
    int arr[] = {5, 1, 4, 2, 8};  
    int N = sizeof(arr) / sizeof(arr[0]);  
  
    bubbleSortAscending(arr, N);  
    cout<<"riya"<<endl;  
    cout<<2210997197<<endl;  
    cout << "Sorted array (ascending): ";  
    printArray(arr, N);  
  
    return 0;  
}
```



| Run | Output  |
|-----|---|
|     | <pre>/tmp/MIT0irN8T2.o riya 2210997197 Sorted array (ascending): 1 2 4 5 8  === Code Execution Successful ===</pre> |

(ii). Create a program to arrange the elements of array in ascending order using Selection Sort algorithm.

```
#include <iostream>
using namespace std;

void swap(int *xp, int *yp) {
    int temp = *xp;
    *xp = *yp;
    *yp = temp;
}

void selectionSort(int arr[], int n) {
    for (int i = 0; i < n - 1; i++) {

        int min_idx = i;
        for (int j = i + 1; j < n; j++) {
            if (arr[j] < arr[min_idx]) {
                min_idx = j;
            }
        }
    }
}
```

```

        swap(&arr[i], &arr[min_idx]);
    }
}

```

```

int main() {
    int arr[] = {64, 25, 12, 22, 11};
    int n = sizeof(arr) / sizeof(arr[0]);
    cout<<"riya"<<endl;
    cout<<2210997197<<endl;
    cout << "Original array: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;

    selectionSort(arr, n);

    cout << "Sorted array: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
    return 0;}

```

| Run | Output  |
|-----|---|
|     | <pre> /tmp/XpXUema9nE.o riya 2210997197 Original array: 64 25 12 22 11 Sorted array: 11 12 22 25 64  === Code Execution Successful === </pre> |

**(iii). Generate a program to arrange the elements of a given array in ascending/descending order**

**using insertion sort**

```
#include <iostream> // Only include necessary headers
```

```
using namespace std;
```

```
void insertionSort(int arr[], int n) {
```

```
    for (int i = 1; i < n; i++) {
```

```
        int key = arr[i];
```

```
        int j = i - 1;
```

```
        while (j >= 0 && arr[j] > key) {
```

```
            arr[j + 1] = arr[j];
```

```
            j--;
```

```
        }
```

```
        arr[j + 1] = key;
```

```
    }
```

```
}
```

```
void printArray(const int arr[], int n) {
```

```
    for (int i = 0; i < n; i++) {
```

```
        cout << arr[i] << " ";
```

```
    }
```

```
    cout << endl;
```

```
}
```

```
int main() {
```

```
    int arr[] = {12, 11, 13, 5, 6};
```

```
    int N = sizeof(arr) / sizeof(arr[0]);
```

```
    cout << "riya" << endl;
```

```
    cout << 2210997197 << endl;
```

```
    cout << "Original array: ";
```

```
    printArray(arr, N);
```

```
    insertionSort(arr, N);
```

```

cout << "Sorted array (ascending order): ";
printArray(arr, N);

return 0;
}

```

| Run | Output   |
|-----|--|
|     | <pre> /tmp/2SU6uHjiJg.o riya 2210997197 Original array: 12 11 13 5 6 Sorted array (ascending order): 5 6 11 12 13  === Code Execution Successful ===S </pre> |

## 5. Generate a code to merge two arrays into a single array

```

#include <iostream>

using namespace std;

int main() {

    // Initialize the first array

    int arr1[] = { 1, 3, 5 };

    int n1 = sizeof(arr1) / sizeof(arr1[0]);

    int arr2[] = { 2, 4, 6 };

    int n2 = sizeof(arr2) / sizeof(arr2[0]);

```

```
int mergedArr[n1 + n2];
```

```
int k = 0;
```

```
for (int i = 0; i < n1; i++) {  
    mergedArr[k++] = arr1[i];  
}
```

```
for (int i = 0; i < n2; i++) {  
    mergedArr[k++] = arr2[i];  
}
```

```
cout<<"riya"<<endl;
```

```
cout<<2210997197<<endl;
```

```
cout << "Merged array: ";
```

```
for (int i = 0; i < n1 + n2; i++) {  
    cout << mergedArr[i] << " ";  
}
```

```
cout << endl;
```

```
return 0;
```

}

| Run | Output   |
|-----|--|
|     | <p data-bbox="512 342 868 383">^ /tmp/KizMSQ24tn.o</p> <p data-bbox="512 398 596 439">riya</p> <p data-bbox="512 454 719 495">2210997197</p> <p data-bbox="512 510 1031 551">Merged array: 1 3 5 2 4 6</p> <p data-bbox="512 667 1203 707">=== Code Execution Successful ===</p> |