1.

```
#include <iostream>

using namespace std;

void arrayInput(int arr[], int n)
{
   for (int i = 0; i < n; i++)
      cin >> arr[i];
}

int search(int arr[], int n, int key)
{
   for (int i = 0; i < n; i++)
   {
      if (key == arr[i])
         return i + 1;
   }
   return 0;
}

void insert(int arr[], int n, int pos, int val)
{
   int i = n - 1;

   while (pos <= i)
   {
      arr[i + 1] = arr[i];
      i--;
   }
   arr[pos] = val;
```

```
}

void del(int arr[], int n, int pos)
{
   int i = pos;

   while (i < n - 1)
   {
      arr[i] = arr[i + 1];
      i++;
   }
}
void print(int arr[], int n)
{
   for (int i = 0; i < n; i++)
      cout << arr[i] << " ";
   cout << endl;
}

int main()
{

   int arr[1000];
   cout << "Enter Size of Array: ";
   int n;
   cin >> n;
   cout << "Enter Elements of the array: " << endl;
   arrayInput(arr, n);

   int op;
```

```
cout << "1. Insert\n2. Delete\n3. Search\n4. Travers\n0. Exit\n";
while (1)
{
   cout << "Enter Option: ";
   cin >> op;
   switch (op)
   {
   case 0:
      return 0;
   case 1:
   {
      cout << "Enter position: ";
      int pos, val;
      cin >> pos;
      cout << "Enter Value: ";
      cin >> val;

      pos--;
      insert(arr, n, pos, val);
      n++;
   }
   break;
   case 2:
   {
      cout << "Enter position: ";
      int pos;
      cin >> pos;
      pos--;
      del(arr, n, pos);
      n--;
   }
```

```
        break;
        case 3:
        {
            cout << "Enter Key: ";
            int key;
            cin >> key;
            int pos = search(arr, n, key);
            if (pos)
                cout << "Your key found at position " << pos << endl;
            else
                cout << "Your key is not found." << endl;
        }
        break;
        case 4:
            print(arr, n);
            break;
        default:
            cout << "You Entered Wrong Option! Try again." << endl;
            break;
        }
    }
}
```

Output:

```
Enter Size of Array: 5
Enter Elements of the array:
8
9
7
3
4
```

1. Insert

2. Delete

3. Search

4. Travers

0. Exit

Enter Option: 1

Enter position: 2

Enter Value: 1

Enter Option: 4

8 1 9 7 3 4

Enter Option: 2

Enter position: 3

Enter Option: 4

8 1 7 3 4

Enter Option: 2

Enter position: 4

Enter Option: 4

8 1 7 4

Enter Option: 3

Enter Key: 7

Your key found at position 3

Enter Option: 0

2.

```
#include <iostream>
using namespace std;
void bubbleSort(string &arr, int n)
{
   char t;
   for (int i = 0; i < n - 1; i++)
   {
      for (int j = 0; j < n - i - 1; j++)
      {
         if (arr[j] > arr[j + 1])
         {
            t = arr[j + 1];
            arr[j + 1] = arr[j];
            arr[j] = t;
         }
      }
   }
}
int main()
{
   cout << "Enter Size of Array: ";
   int n;
   cin >> n;
   string arr;
   cout << "Enter String: " << endl;
   cin >> arr;
   bubbleSort(arr, n);
   cout << "Sorted Array" << endl;
   for (int i = 0; i < n; i++)
   {
```

```
        cout << arr[i] << " ";
    }
    cout << endl;
    return 0;
}
```

Output:

```
Enter Size of Array: 6
Enter String:
PeopLE
Sorted Array
E L P e o p
```

3.

```
#include <bits/stdc++.h>

using namespace std;

void arrayInput(int arr[], int n)
{
    for (int i = 0; i < n; i++)
        cin >> arr[i];
}

void print(int arr[], int n)
{
    for (int i = 0; i < n; i++)
        cout << arr[i] << " ";
    cout << endl;
}

int binarySearch(int arr[], int l, int h, int key)
{
    if (l > h)
        return 0;

    int mid = (l + h) / 2;

    if (arr[mid] == key)
        return mid + 1;

    if (arr[mid] < key)
        return binarySearch(arr, mid + 1, h, key);
    else
```

```
        return binarySearch(arr, l, mid - 1, key);
}

int main()
{
   cout << "Enter Size of Array: ";
   int n;
   cin >> n;
   int arr[n];
   cout << "Enter Elements of the array: " << endl;
   arrayInput(arr, n);
   sort(arr, arr + n);
   int key;
   cout << "Enter Key: ";
   cin >> key;
   int pos = binarySearch(arr, 0, n - 1, key);
   if (pos)
      cout << "Your key found." << endl;
   else
      cout << "Your key is not found." << endl;
   return 0;
}
```

Output:

```
Enter Size of Array: 4
Enter Elements of the array:
8
4
6
9
Enter Key: 6
Your key found.
```