

Excesice 01:

1.bubble sort &insertion sort

1.bubble sort

```
void bubbleSort(vector<int> &a)
{
    for (int i = 0; i < a.size(); i++)
    {
        for (int j = a.size() - 1; j > i; j--)
        {
            if (a[j] < a[j-1])
            {
                int tam = a[j];
                a[j] = a[j-1];
                a[j-1] = tam;
                print_list(a);
            }
        }
        cout << "Sort by bubble sort:\n";
        print_list(a);
    }
}
```

```
Input n:5
1. Bubble Sort
2.Insertion Sort
You choose: 1
45 -5 98 -10 0
45 -5 -10 98 0
45 -10 -5 98 0
-10 45 -5 98 0
-10 45 -5 0 98
-10 -5 45 0 98
-10 -5 0 45 98
Sort by bubble sort:
-10 -5 0 45 98
```

2. Insertion sort

```

✓ void InsertionSort(vector<int> &a)
{
    int x, j;
    ✓ for (int i = 1; i < a.size(); i++)
    {
        x=a[i];
        j = i-1;
        ✓ while (0<=j && x<a[j])
        {
            a[j+1]=a[j];
            j--;
        }
        a[j+1]=x;
        print_list(a);
    }
    cout <<"Sort by insertion sort:\n";
    print_list(a);
}

```

Input your array:-4 56 0 7 -1

Your array is:

-4 56 0 7 -1

1. Bubble Sort

2. Insertion Sort

You choose: 2

-4 56 0 7 -1

-4 56 0 7 -1

-4 0 56 7 -1

-4 0 7 56 -1

-4 -1 0 7 56

Sort by insertion sort:

-4 -1 0 7 56

Thanks for using!

2. Quick sort

```

void QuickSort(vector<int> &b,int left, int right,int n)
{
    int x = b[(left+right)/2];
    int i=left;
    int j =right;
    while (i<j)
    {
        while (b[i]<x)
        {
            i++;
        }
        while (b[j]>x)
        {
            j--;
        }
        if (i<=j)
        {
            {
                int tam= b[i];
                b[i]=b[j];
                b[j]=tam;
            }
            //swap(b[i],b[j]);

            i++;
            j--;
        }
        if (left <j) QuickSort(b,left,j,b.size());
        if (i<right) QuickSort(b,i,right,b.size());
    }
    // print_list(a);
}

```

Input n:5

Input your array:4 90 -3 0 -10

Your array is:

4 90 -3 0 -10

-10 -3 0 4 90