

Jobsheet 8  
Praktikum Struktur Data

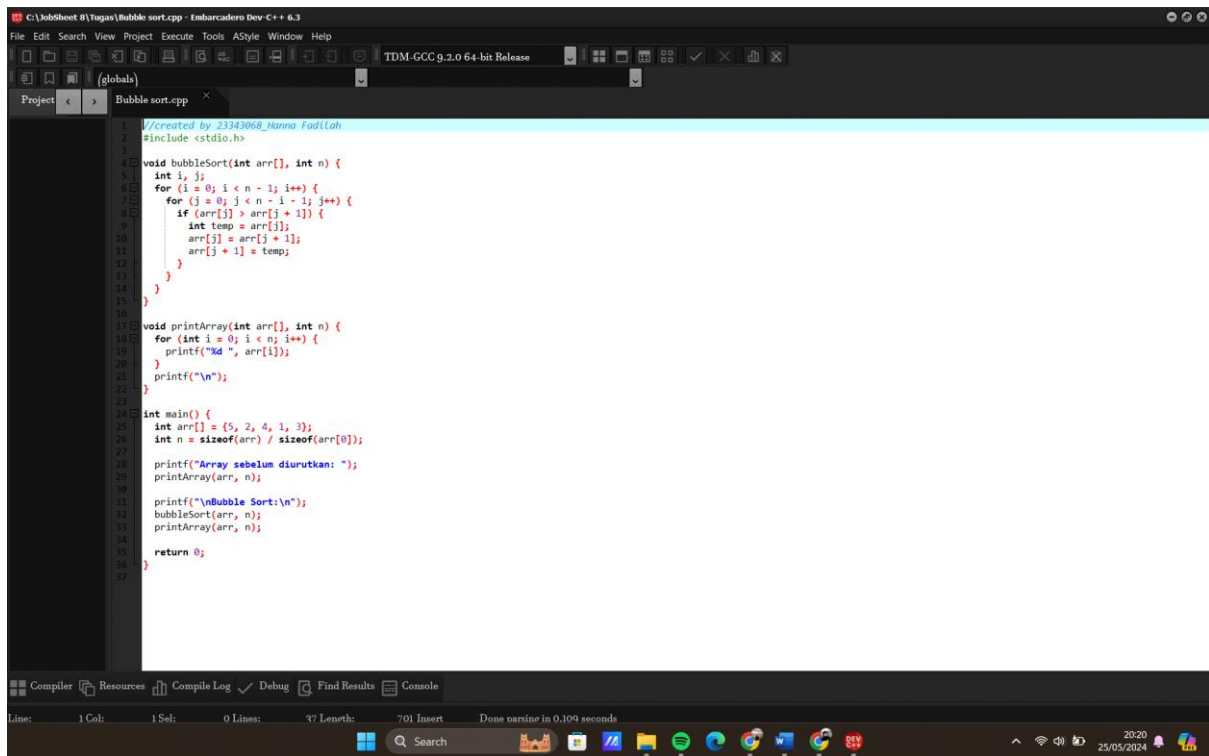


Dosen pengampu : Randi Proska Sandra, S.Pd, M.Sc  
Kode Kelas : 202323430158

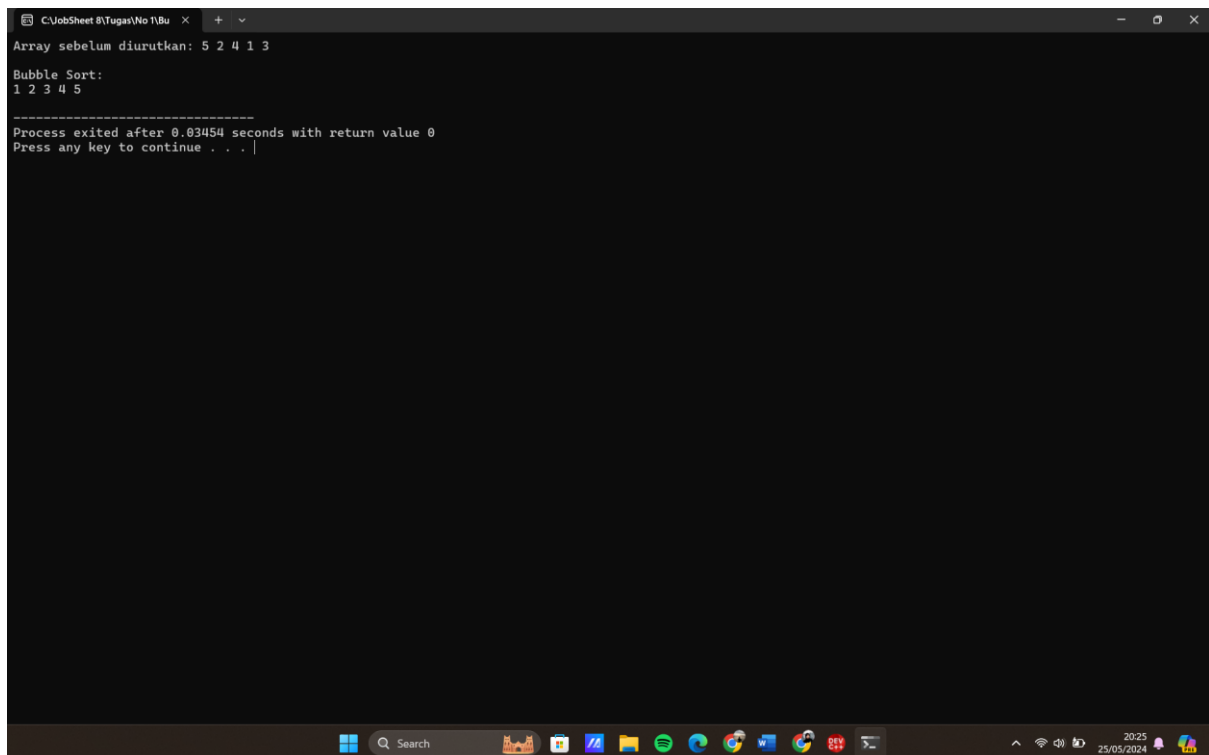
Disusun Oleh :  
Hanna Fadilah  
23343068

PROGRAM STUDI INFORMATIKA (NK)  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI PADANG  
2024

# 1. Bubble Sort



```
1 //created by 23343068_Hanna Fadillah
2 #include <stdio.h>
3
4 void bubbleSort(int arr[], int n) {
5     int i, j;
6     for (i = 0; i < n - 1; i++) {
7         for (j = 0; j < n - i - 1; j++) {
8             if (arr[j] > arr[j + 1]) {
9                 int temp = arr[j];
10                arr[j] = arr[j + 1];
11                arr[j + 1] = temp;
12            }
13        }
14    }
15
16    void printArray(int arr[], int n) {
17        for (int i = 0; i < n; i++) {
18            printf("%d ", arr[i]);
19        }
20        printf("\n");
21    }
22
23    int main() {
24        int arr[] = {5, 2, 4, 1, 3};
25        int n = sizeof(arr) / sizeof(arr[0]);
26
27        printf("Array sebelum diurutkan: ");
28        printArray(arr, n);
29
30        printf("\nBubble Sort:\n");
31        bubbleSort(arr, n);
32        printArray(arr, n);
33
34        return 0;
35    }
36 }
```



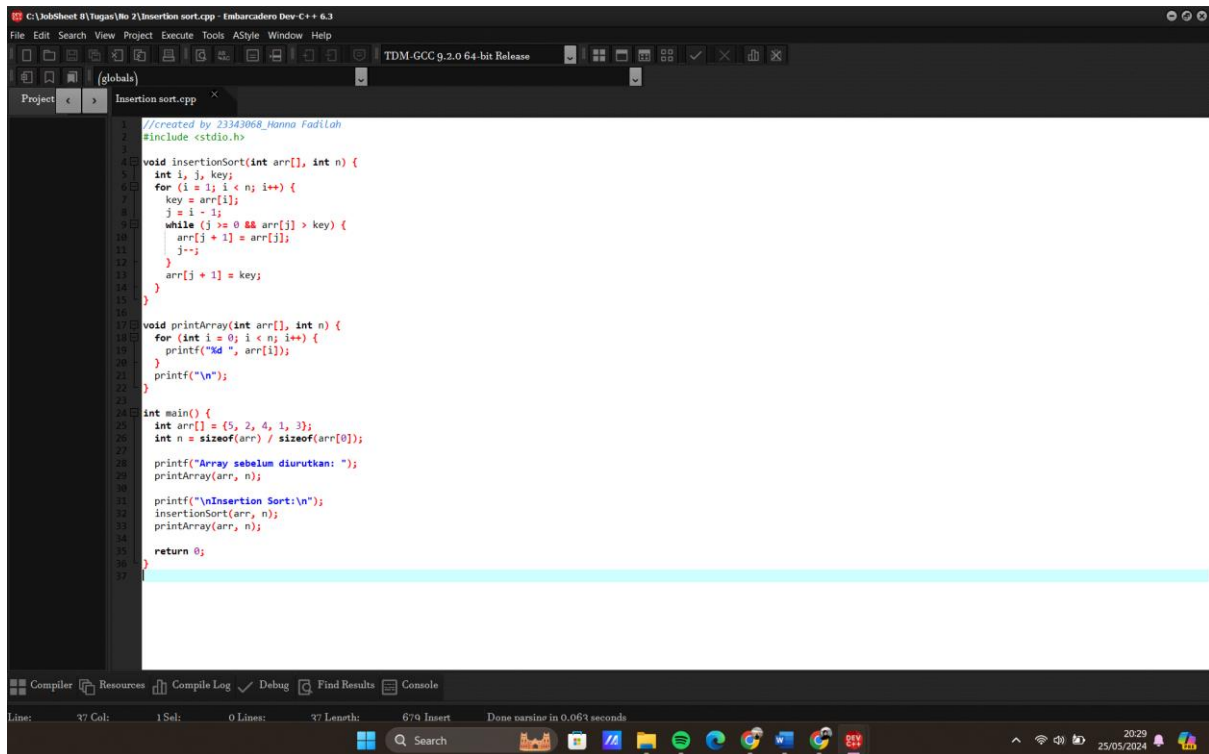
```
C:\JobSheet 8\Tugas\No 1\Bu x + v
Array sebelum diurutkan: 5 2 4 1 3
Bubble Sort:
1 2 3 4 5

-----
Process exited after 0.03454 seconds with return value 0
Press any key to continue . . . |
```

## Penjelasan

- Algoritma pengurutan sederhana yang mengurutkan data dari kecil ke besar (ascending) atau besar ke kecil (descending).
- Bekerja dengan cara membandingkan dua elemen yang berdekatan dan menukar posisinya jika urutannya salah.
- Diulang hingga semua elemen berada pada urutan yang benar.

## 2. Insertion Sort

A screenshot of a C++ IDE window titled "C:\Jobsheet 8\Tugas\No 2\Insertion sort.cpp - Embarcadero Dev-C++ 6.3". The code implements the Insertion Sort algorithm. It includes a header file, defines a function to insert an element into a sorted array, a function to print the array, and a main function that tests the algorithm with an array of five numbers. The IDE interface shows the code editor, a project pane on the left, and a status bar at the bottom indicating 37 lines of code and a successful compilation.

```
1 //created by 23343068_Hanna Fadilah
2 #include <stdio.h>
3
4 void insertionSort(int arr[], int n) {
5     int i, j, key;
6     for (i = 1; i < n; i++) {
7         key = arr[i];
8         j = i - 1;
9         while (j >= 0 && arr[j] > key) {
10             arr[j + 1] = arr[j];
11             j--;
12         }
13         arr[j + 1] = key;
14     }
15 }
16
17 void printArray(int arr[], int n) {
18     for (int i = 0; i < n; i++) {
19         printf("%d ", arr[i]);
20     }
21     printf("\n");
22 }
23
24 int main() {
25     int arr[] = {5, 2, 4, 1, 3};
26     int n = sizeof(arr) / sizeof(arr[0]);
27
28     printf("Array sebelum diurutkan: ");
29     printArray(arr, n);
30
31     printf("\nInsertion Sort:\n");
32     insertionSort(arr, n);
33     printArray(arr, n);
34
35     return 0;
36 }
```

```
C:\JobSheet 8\Tugas\No 2\Un >
Array sebelum diurutkan: 5 2 4 1 3
Insertion Sort:
1 2 3 4 5

-----
Process exited after 0.03304 seconds with return value 0
Press any key to continue . . .
```

## Penjelasan

- Algoritma pengurutan yang menyisipkan elemen satu per satu ke dalam urutan yang benar.
- Bekerja seperti menyusun kartu remi di tangan.
- Memiliki kompleksitas waktu rata-rata  $O(n)$  dan terburuk  $O(n^2)$ .