

# **Tugas Besar Algoritma dan Struktur Data**



Nama:

Firja Rakha Adwitty / 4522210072

**PROGRAM STUDI TEKNIK INFORMATIKA**

**UNIVERSITAS PANCASILA**

**2023**

```

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

// Shell Sort
void shellSort(vector<int>& arr) {
    int n = arr.size();

    for (int gap = n / 2; gap > 0; gap /= 2) {
        for (int i = gap; i < n; i++) {
            int temp = arr[i];
            int j;
            for (j = i; j >= gap && arr[j - gap] > temp; j -= gap) {
                arr[j] = arr[j - gap];
            }
            arr[j] = temp;
        }
    }
}

// Bubble Sort
void bubbleSort(vector<int>& arr) {
    int n = arr.size();
    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]);
            }
        }
    }
}

// Merge Sort
void merge(vector<int>& arr, int left, int mid, int right) {
    int n1 = mid - left + 1;
    int n2 = right - mid;

    vector<int> L(n1), R(n2);

    for (int i = 0; i < n1; i++)
        L[i] = arr[left + i];

```

```

    for (int j = 0; j < n2; j++)
        R[j] = arr[mid + 1 + j];

    int i = 0, j = 0, k = left;

    while (i < n1 && j < n2) {
        if (L[i] <= R[j]) {
            arr[k] = L[i];
            i++;
        }
        else {
            arr[k] = R[j];
            j++;
        }
        k++;
    }

    while (i < n1) {
        arr[k] = L[i];
        i++;
        k++;
    }

    while (j < n2) {
        arr[k] = R[j];
        j++;
        k++;
    }
}

void mergeSort(vector<int>& arr, int left, int right) {
    if (left < right) {
        int mid = left + (right - left) / 2;

        mergeSort(arr, left, mid);
        mergeSort(arr, mid + 1, right);

        merge(arr, left, mid, right);
    }
}

```

```

// Quick Sort
int partition(vector<int>& arr, int low, int high) {
    int pivot = arr[high];
    int i = (low - 1);

    for (int j = low; j <= high - 1; j++) {
        if (arr[j] < pivot) {
            i++;
            swap(arr[i], arr[j]);
        }
    }
    swap(arr[i + 1], arr[high]);
    return (i + 1);
}

void quickSort(vector<int>& arr, int low, int high) {
    if (low < high) {
        int pi = partition(arr, low, high);

        quickSort(arr, low, pi - 1);
        quickSort(arr, pi + 1, high);
    }
}

// Function to print an array
void printArray(const vector<int>& arr) {
    int n = arr.size();
    for (int i = 0; i < n; i++)
        cout << arr[i] << " ";
    cout << endl;
}

int main() {
    cout << "\n-----" << endl;
    cout << " Nama : Firja Rakha Adwitty " << endl;
    cout << " NIM : 4522210072 " << endl;
    cout << "-----" << endl;
    cout << " Tugas Besar Praktikum " << endl;
    cout << "-----" << endl;

    vector<int> arr = { 64, 34, 25, 12, 22, 11, 90 };
}

```

```

int n = arr.size();
int pilihan = 0;

while (pilihan != 5) {
    cout << "\n===== " <<
endl;
    cout << "Data sebelum diurutkan : " << endl;
    printArray(arr);
    cout << "\n";
    cout << " Menu\n";
    cout << " 1. Shell Sort" << endl;
    cout << " 2. Bubble Sort" << endl;
    cout << " 3. Merge Sort" << endl;
    cout << " 4. Quick Sort" << endl;
    cout << " 5. Keluar" << endl;
    cout << "\nPilih salah satu algoritma Sorting (1 sampai 5): ";
    cin >> pilihan;

    switch (pilihan) {
    case 1:
        shellSort(arr);
        cout << "\nHasil Pengurutan dengan Shell Sort: \n";
        printArray(arr);
        cout << "SELESAI" << endl;
        break;
    case 2:
        bubbleSort(arr);
        cout << "\nHasil Pengurutan dengan Bubble Sort: \n";
        printArray(arr);
        cout << "SELESAI" << endl;
        break;
    case 3:
        mergeSort(arr, 0, n - 1);
        cout << "\nHasil Pengurutan dengan Merge Sort: \n";
        printArray(arr);
        cout << "SELESAI" << endl;
        break;
    case 4:
        quickSort(arr, 0, n - 1);
        cout << "\nHasil Pengurutan dengan Quick Sort: \n";
        printArray(arr);

```

```
        cout << "SELESAI" << endl;
        break;
    }
}

return 0;
}

// This is code is contributed by rathbhupendra
```

```
C:\Users\ASUS\OneDrive\Documents\Coding Vs Code\Semester 2\Juli 2023>g++ Tugas_Besar_4522210072_Firja_Rakha.cpp -o Tugas_Besar_4522210072_Firja_Rakha.exe
```

```
C:\Users\ASUS\OneDrive\Documents\Coding Vs Code\Semester 2\Juli 2023>Tugas_Besar_4522210072_Firja_Rakha
```

```
=====
Nama : Firja Rakha Adwittya
NIM : 4522210072
=====
```

```
Tugas Besar Praktikum
=====
```

```
=====
Data sebelum diurutkan :
64 34 25 12 22 11 90
```

```
Menu
1. Shell Sort
2. Bubble Sort
3. Merge Sort
4. Quick Sort
5. Keluar
```

Pilih salah satu algoritma Sorting (1 sampai 5): 1

```
Hasil Pengurutan dengan Shell Sort:
11 12 22 25 34 64 90
SELESAI
```

```
=====
Data sebelum diurutkan :
11 12 22 25 34 64 90
```

```
Menu
1. Shell Sort
2. Bubble Sort
3. Merge Sort
4. Quick Sort
5. Keluar
```

Pilih salah satu algoritma Sorting (1 sampai 5): 2

```
Hasil Pengurutan dengan Bubble Sort:
11 12 22 25 34 64 90
SELESAI
```

```
=====
Data sebelum diurutkan :
11 12 22 25 34 64 90
```

```
Menu
1. Shell Sort
2. Bubble Sort
3. Merge Sort
4. Quick Sort
5. Keluar
```

Pilih salah satu algoritma Sorting (1 sampai 5): 3

```
Hasil Pengurutan dengan Merge Sort:
11 12 22 25 34 64 90
SELESAI
```

```
=====
Data sebelum diurutkan :
11 12 22 25 34 64 90
```

```
Menu
1. Shell Sort
2. Bubble Sort
3. Merge Sort
4. Quick Sort
5. Keluar
```

Pilih salah satu algoritma Sorting (1 sampai 5): 4

```
Hasil Pengurutan dengan Quick Sort:
11 12 22 25 34 64 90
SELESAI
```

```
=====
Data sebelum diurutkan :
11 12 22 25 34 64 90
```

```
Menu
1. Shell Sort
2. Bubble Sort
3. Merge Sort
4. Quick Sort
5. Keluar
```

Pilih salah satu algoritma Sorting (1 sampai 5): 5

```
C:\Users\ASUS\OneDrive\Documents\Coding Vs Code\Semester 2\Juli 2023>|
```