

Algoritma & Struktur Data
M11-Sorting 2 (bubble & shell)

Dosen Pengampu

Dr. Tita Karlita S.Kom, M.Kom



Disusun Oleh :

Nama : M. Faza Nur Husain

Nrp : 3121550004

**D3 PJJ AK TEKNIK INFORMATIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
TAHUN AKADEMIK 2021/2022**

1. Bubble Sort

Source Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define MAX 10

time_t t1, t2;
long waktuKomputasi;

void printArray(int data[])
{
    for (int i = 0; i < MAX; i++)
    {
        printf("%d ", data[i]);
    }
    puts("");
}

void randArray(int data[])
{
    for (int i = 0; i < MAX; i++)
    {
        data[i] = rand() / 3000 + 1;
        printf("%d ", data[i]);
    }
    puts("");
}

void bubbleSort(int data[])
{
    int did_swap = 1;
    int i, j;
    int temp;
    int banding = 0, tukar = 0, geser = 0;

    for (i = 0; i < MAX; i++)
    {
        if (did_swap == 1)
        {
            did_swap = 0;
            for (j = 0; j < MAX - i - 1; j++)
            {
                banding = banding + 1;
                ;
                if (data[j] > data[j + 1])
                {
                    tukar = tukar + 1;
                    temp = data[j];
                    data[j] = data[j + 1];
                    data[j + 1] = temp;
                    geser = geser + 3;
                    did_swap = 1;
                }
            }
        }
    }
    printArray(data);
    puts("");
    printf("Jumlah Perbandingan %d \n", banding);
    printf("Jumlah Pertukaran %d \n", tukar);
    printf("Jumlah Pergeseran %d \n", geser);
}

int main()
{

```

```
int nilai[MAX];
// int nilai[MAX] = {3, 5, 1, 9, 2, 0, 7, 8, 6, 4};
printf("--- BUBBLE SORT ---\n");
puts("");
printf("Sebelum Diurutkan : ");
randArray(nilai);
// printArray(nilai);
printf("Setelah Diurutkan : ");
time(&t1);
bubbleSort(nilai);
time(&t2);
waktuKomputasi = t2 - t1;

puts("");
printf("Waktu Komputasi = %g\n", waktuKomputasi);
return 0;
}
```

Output Source Code :

```
--- BUBBLE SORT ---

Sebelum Diurutkan : 1 7 3 9 7 6 4 10 9 9
Setelah Diurutkan : 1 3 4 6 7 7 9 9 9 10

Jumlah Perbandingan 35
Jumlah Pertukaran 11
Jumlah Pergeseran 33

Waktu Komputasi = 8.00858e-307

Process returned 0 (0x0)   execution time : 0.011 s
Press any key to continue.
|
```

2. Shell Sort

Source Code :

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define MAX 10

time_t t1, t2;
long waktuKomputasi;

void printArray(int data[])
{
    for (int i = 0; i < MAX; i++)
    {
        printf("%d ", data[i]);
    }
    printf("\n");
}

void shellSort(int data[])
{
    int did_swap = 1;
    int i, jarak;
    int temp;
    int banding = 0, tukar = 0, geser = 0;

    jarak = MAX;
    sleep(3);
    while (jarak > 1)
    {
        jarak /= 2;
        did_swap = 1;
        while (did_swap == 1)
        {
            did_swap = 0;
            i = 0;
            while (i < (MAX - jarak))
            {
                banding++;
                if (data[i] > data[i + jarak])
                {
                    tukar++;
                    temp = data[i];
                    data[i] = data[i + jarak];
                    data[i + jarak] = temp;
                    did_swap = 1;
                }
                i++;
            }
        }
    }

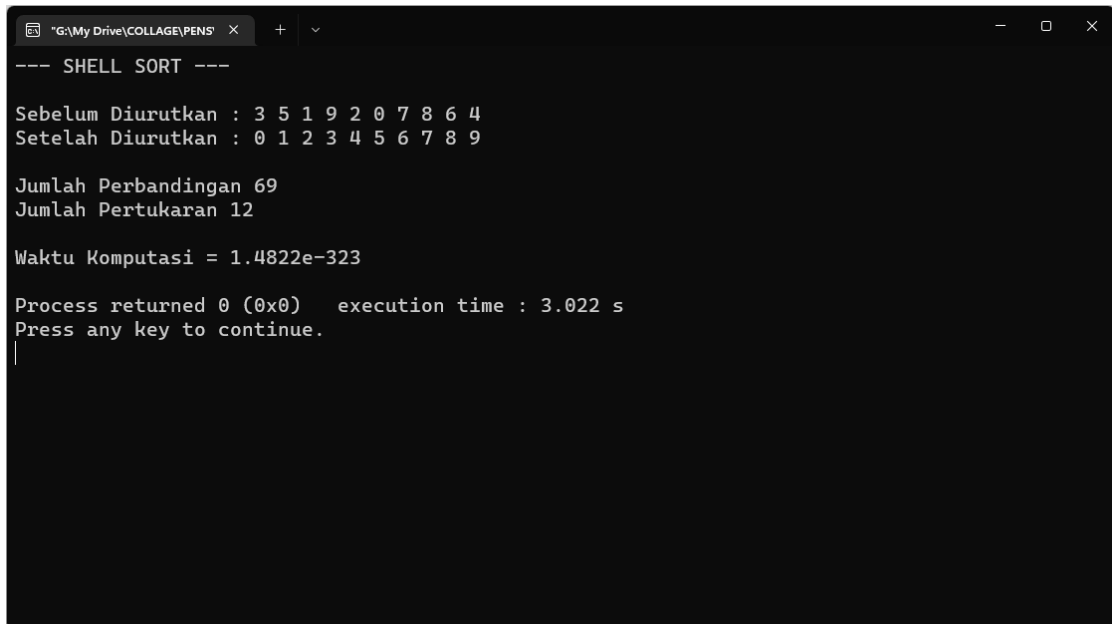
    printArray(data);
    puts("");
    printf("Jumlah Perbandingan %d \n", banding);
    printf("Jumlah Pertukaran %d \n", tukar);
}

int main()
{
    int nilai[MAX] = {3, 5, 1, 9, 2, 0, 7, 8, 6, 4};
    printf("--- SHELL SORT ---\n");
    puts("");
    printf("Sebelum Diurutkan : ");
    printArray(nilai);
    printf("Setelah Diurutkan : ");
}
```

```
time(&t1);
shellSort(nilai);
time(&t2);
waktuKomputasi = t2 - t1;
// printArray(nilai);

puts("");
printf("Waktu Komputasi = %g\n", waktuKomputasi);
return 0;
}
```

Output Source Code :



```
--- SHELL SORT ---

Sebelum Diurutkan : 3 5 1 9 2 0 7 8 6 4
Setelah Diurutkan : 0 1 2 3 4 5 6 7 8 9

Jumlah Perbandingan 69
Jumlah Pertukaran 12

Waktu Komputasi = 1.4822e-323

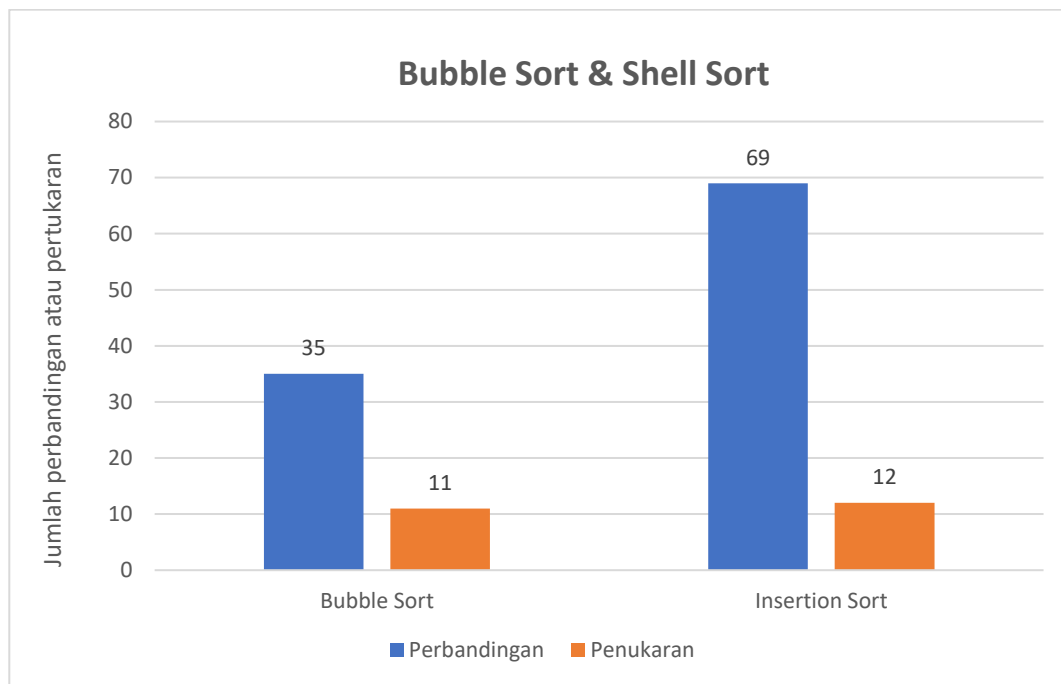
Process returned 0 (0x0)   execution time : 3.022 s
Press any key to continue.
```

3. Analisa terhadap no 1 dan 2. terkait jumlah perbandingan, jumlah penukaran dan waktu komputasi.

a) Grafik perbandingan untuk masing2 jumlah perbandingan, jumlah penukaran.

b)

	Bubble Sort	Shell Sort
Perbandingan	35	69
Penukaran	11	12



c) Grafik perbandingan untuk masing2 jumlah waktu komputasi.

	Bubble Sort	Shell Sort
Waktu komputasi	8,00858e-307	1,4822e-323

