



Modul Praktikum

Pemrograman Lanjut

Laboratium Digital



Tim Penyusun Modul

Tim Asisten Lab Digital 2021

Program Studi S1 Teknik Komputer

Fakultas Teknik

Universitas Indonesia

2021

KATA PENGANTAR

Praktikum ini adalah bagian dari mata kuliah Pemrograman Lanjut dan Praktikum yang diberikan untuk mahasiswa Semester 2 Program Studi Teknik Komputer. Mata kuliah ini merupakan mata kuliah lanjutan yang diberikan kepada mahasiswa dari serangkaian mata kuliah untuk mendukung 2 CP Prodi, yaitu Mampu merancang algoritma untuk masalah tertentu dan mengimplementasikannya ke dalam pemrograman (C6) dan Mampu memanfaatkan teknologi informasi dan komunikasi (C3).

Capaian Pembelajaran Mata Kuliah (CPMK) dari mata kuliah ini adalah:

1. Mampu merancang program komputer prosedural kompleks dengan struktur data dinamis (C6)
2. Mampu menunjukkan sikap kritis, kreatif, dan inovatif dan menghargai orang lain dalam kelompok untuk memecahkan masalah bersama dan tugas kelompok Pemrograman Lanjut (C3, A3)
3. Mampu menggunakan software pemrogram komputer dengan mahir (C3)

Sedangkan Sub-CPMK yang akan dicapai adalah:

- 1.1. Mampu mengimplementasikan algoritma rekursif ke dalam pemrograman
- 1.2. Mampu mengimplementasikan algoritma searching dan sorting ke dalam pemrograman
- 1.3. Mampu membuat program komputer prosedural kompleks dengan linked list, stack dan queue
- 1.4. Mampu mengimplementasikan multi-threading dan parallel programming
- 1.5. Mampu merancang perangkat lunak sederhana dengan struktur data dinamis
- 2.1. Mampu menunjukkan proses berpikir kritis, kreatif dan inovatif dalam menyelesaikan permasalahan kelompok
- 2.2. Mampu berkomunikasi dengan sopan
- 2.3. Mampu menghargai pendapat orang lain
- 3.1. Mampu menggunakan software pemrogram komputer untuk program kompleks dengan mahir

Pada mata kuliah ini mahasiswa akan mengasah kemampuan cara berpikir dan penyelesaian masalah dengan membuat algoritma, kemudian menerjemahkan algoritma tersebut ke dalam bahasa pemrograman yang dapat dijalankan oleh komputer. Pada mata kuliah ini mahasiswa akan mempelajari Rekursif, Searching, Sorting, Linked list, Stack, Queue, Multi-threading, Parallel programming,, dan pada bagian akhir akan ditutup dengan proyek akhir pemrograman yang dibuat oleh mahasiswa.

Bahasa C merupakan bahasa pemrograman terstruktur, yang membagi program dalam sejumlah blok. Tujuannya adalah untuk mempermudah dalam pembuatan dan pengembangan program. Bahasa C menggunakan standarisasi ANSI (American National Standardization Institute) yang dijadikan acuan oleh para pembuat compiler C. Bahasa C terdiri dari fungsi-fungsi dan setiap program C memiliki fungsi utama yang disebut main. Program akan dieksekusi dimulai dari statement pertama pada fungsi main tersebut.

Akhir kata, diharapkan modul praktikum ini akan dapat menjadi referensi untuk membuat program dalam Bahasa C secara umum, dan menjadi panduan dalam menjalankan praktikum mata kuliah Pemrograman Lanjut, secara khusus.

Depok, Februari 2021

Tim Penyusun Modul

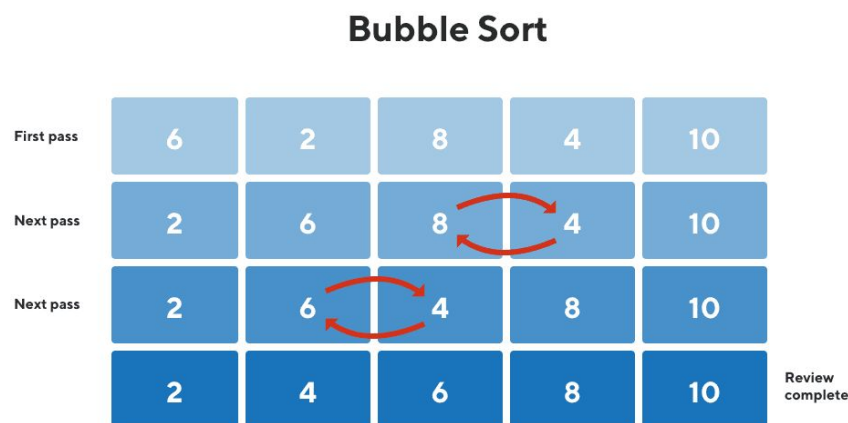
MODUL 4: PENGURUTAN

Melakukan pengurutan data baik secara *ascending* maupun *descending* merupakan pengaplikasian pemrograman yang penting, karena pada kenyataanya data yang dikumpulkan perlu diurutkan terhadap variabel tertentu.

Pada modul ini kita akan membahas beberapa metode sorting seperti **Bubble sort**, **Selection sort**, dan **Insertion sort**.

Bubble Sorting

Jenis sorting ini merupakan jenis pengurutan paling sederhana, dimana prosesnya dilakukan dengan melakukan pengecekan satu persatu secara berulang kali dan menggantinya jika berada pada urutan yang salah.



Dapat dilihat pada gambar diatas, pada proses looping program akan mengecek angka yang terdapat dalam array. Jika terdapat urutan yang salah maka algoritma program tersebut akan menukar posisi dari isi array tersebut. Proses looping tersebut akan berjalan terus hingga tidak terjadi pertukaran sama sekali dalam array.

Contoh Program yang menggunakan Bubble Sorting :

```
#include <stdio.h>
#define SIZE 10

void show(int arr[])
{
```

```

    int i;
    for(i=0; i<SIZE; i++){
        printf("%4d", arr[i]);
    }
    printf("\n");
}

int main(void)
{
    int arr[SIZE] = {10, 2, 6, 34, 5, 25, 42, 12, 76, 8};
    int i,j,temp;

    printf("Isi Array Sebelum Diurutkan : \n");
    show(arr);

    //Lakukan Bubble Sorting
    //Looping untuk setiap isi array
    for(i=0; i<SIZE - 1 ; i++){
        int swapped = 0;
        //Looping untuk perbandingan
        for(j=0; j< SIZE - i - 1 ; j++){
            //ganti operator perbandingan jika
            //ingin pengurutan secara terbalik
            if(arr[j] > arr[j+1]){
                temp = arr[j];
                arr[j] = arr[j+1];
                arr[j+1] = temp;
                //Menandakan terjadi pertukaran
                swapped = 1;
            }

        }
        if(swapped == 0)
            break;
    }

    printf("Isi Array setelah diurutkan : \n");
    show(arr);
}

```

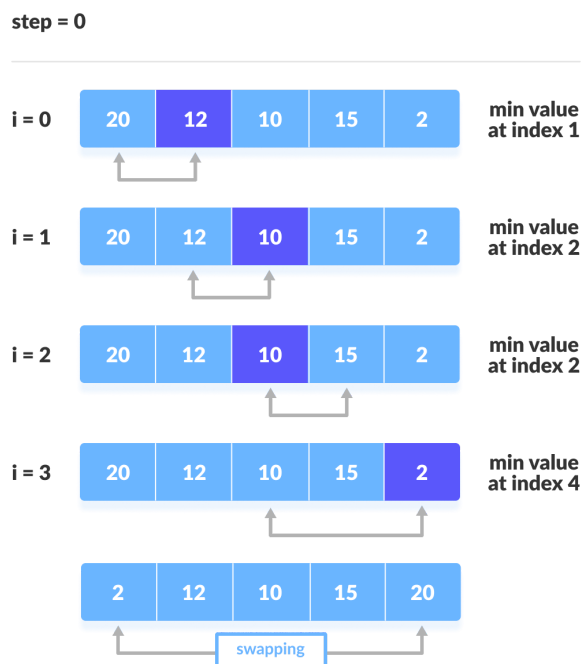
Output Program :

```
"C:\Users\Dar\Desktop\Modul Progran\bin\Debug\Reviewing.exe"
Isi Array Sebelum Diurutkan :
 10  2  6 34  5 25 42 12 76  8
Isi Array setelah diurutkan :
  2  5  6  8 10 12 25 34 42 76

Process returned 0 (0x0)   execution time : 0.039 s
Press any key to continue.
```

Selection Sorting

Jenis algoritma sorting ini dilakukan dengan mencari elemen terkecil pada array lalu menukarnya dengan posisi pertama. Kemudian pada looping selanjutnya program akan mencari elemen terkecil lagi pada array dan menukarnya dengan posisi kedua. Proses tersebut berlanjut hingga urutan array sesuai yang diinginkan (*ascending* atau *descending*)



Contoh program yang menggunakan Selection sort :

```
#include <stdio.h>
#define SIZE 10
```

```

void show(int arr[])
{
    int i;
    for(i=0; i<SIZE; i++){
        printf("%4d", arr[i]);
    }
    printf("\n");
}

int main()
{
    int arr[SIZE] = {10, 2, 6, 34, 5, 25, 42, 12, 76, 8};
    int i, j, position, temp;

    printf("Isi Array Sebelum Diurutkan : \n");
    show(arr);

    for(i=0; i < SIZE - 1; i++){
        position = i;

        //Mencari Nilai terkecil dalma array
        for(j=i+1; j<SIZE; j++){
            if(arr[position] > arr[j])
                position = j;
        }

        //Menukar posisi array
        if(position != i){
            temp = arr[i];
            arr[i] = arr[position];
            arr[position] = temp;
        }
    }

    printf("Isi Array setelah diurutkan : \n");
    show(arr);
}

```

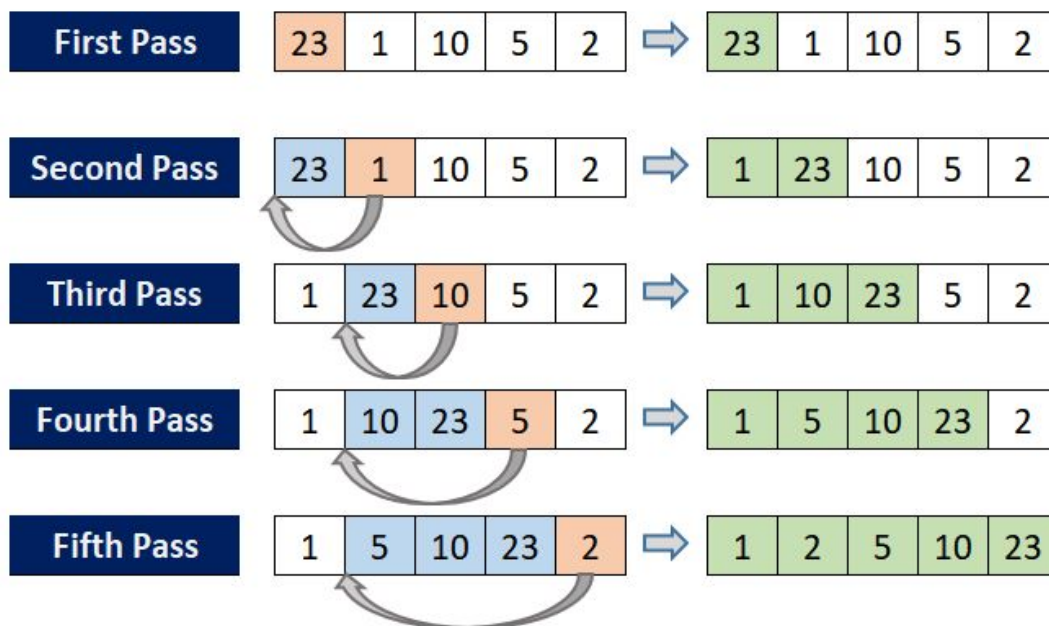

Output Program :

```
"C:\Users\Dar\Desktop\Modul Progran\bin\Debug\Reviewi...
Isi Array Sebelum Diurutkan :
10  2  6 34  5 25 42 12 76  8
Isi Array setelah diurutkan :
 2  5  6 8 10 12 25 34 42 76

Process returned 0 (0x0)   execution time : 0.012 s
Press any key to continue.
```

Insertion Sorting

Metode insertion sort dilakukan dengan menganggap terdapat 2 bagian pada array, yaitu bagian yang sudah terurut dan yang belum terurut. Kemudian setiap elemen pada bagian yang belum terurut diambil satu persatu dan diletakkan pada posisi yang sesuai di bagian yang sudah terurut.



Contoh Program yang menggunakan Insertion Sort :

```
#include <stdio.h>
#define SIZE 10
```

```

void show(int arr[])
{
    int i;
    for(i=0; i<SIZE; i++){
        printf("%4d", arr[i]);
    }
    printf("\n");
}

int main(void)
{
    int arr[SIZE] = {10, 2, 6, 34, 5, 25, 42, 12, 76, 8};
    int i, j, key;

    printf("Isi Array Sebelum Diurutkan : \n");
    show(arr);

    for(i=0; i<SIZE; i++){
        key = arr[i];
        j = i - 1;

        //Looping untuk memindahkan elemen array
        //dengan nilai yang lebih besar dari key
        while (j >= 0 && arr[j] > key) {
            arr[j+1] = arr[j];
            j = j - 1;
        }
        arr[j + 1] = key;
    }

    printf("Isi Array setelah diurutkan : \n");
    show(arr);
}

```

Output Program :

```
"C:\Users\Dar\Desktop\Modul Proglan\bin\Debug\..."
Isi Array Sebelum Diurutkan :
 10  2  6 34  5 25 42 12 76  8
Isi Array setelah diurutkan :
  2  5  6  8 10 12 25 34 42 76

Process returned 0 (0x0)   execution time : 0.033 s
Press any key to continue.
```