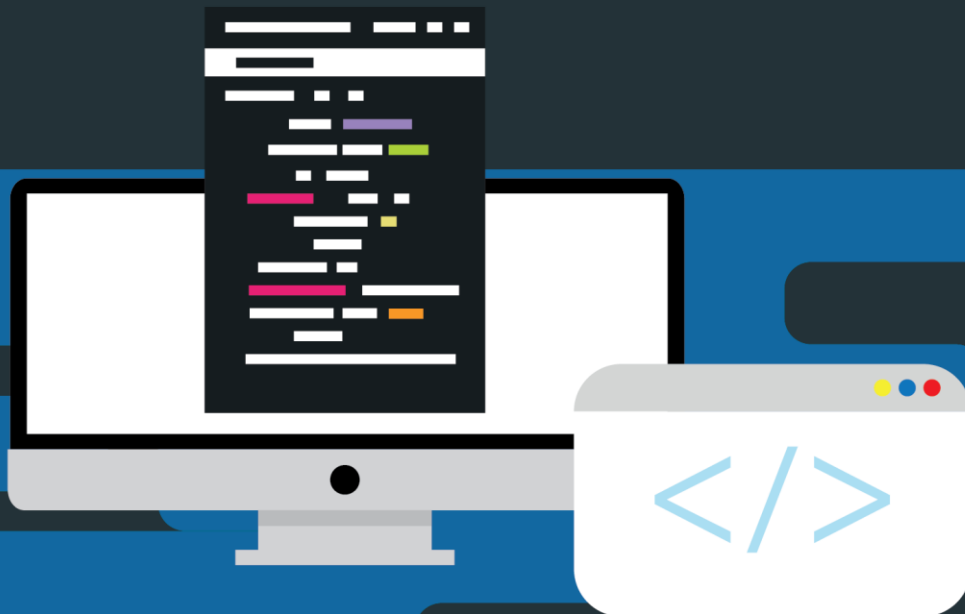


**MODUL PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN II**

SEARCHING



**TIM ASISTEN PEMROGRAMAN
ANGKATAN 11
ILMU KOMPUTER FPMIPA UPI**

SEARCHING

Searching adalah mencari suatu data dalam kumpulan data.

SEQUENTIAL SEARCH

Sequential Search adalah pencarian yang dilakukan dengan mengecek semua data yang ada hingga menemukan data yang dicari. Kelebihan dari search ini adalah data yang dicari selalu ditemukan. Kekurangannya adalah jika data sangat banyak, maka tidak efisien untuk mengecek data satu per satu.

Berikut contoh penggunaan Sequential Search :

```

void sequenSearch(int arr[], int n, int cari){

    int i, found;
    found = 0;
    i = 0;

    while((i < n) && (found == 0)){
        if(arr[i] == cari){
            found = 1;
        }
        else{
            i++;
        }
    }

    if(found == 1){
        printf("Angka %d ditemukan.\n", cari);
    }
    else{
        printf("Data tidak ditemukan.\n");
    }
}

int main(){

    int n, cari;
    printf("Masukkan jumlah data : ");
    scanf("%d", &n);
    int arr[n];

    int i;
    printf("Masukkan data : \n");
    for(i = 0; i < n; i++){
        scanf("%d", &arr[i]);
    }

    printf("Masukkan data yang akan dicari : ");
    scanf("%d", &cari);

    sequenSearch(arr, n, cari);

    return 0;
}

```

BINARY SEARCH

Binary Search adalah pencarian yang dilakukan pada data yang terurut dengan membagi sub data menjadi 2 bagian. Kelebihannya adalah lebih efisien dari Sequential. Kekurangannya tidak semua kasus bisa diselesaikan dengan Binary Search.

Berikut contoh penggunaan Binary Search :

```

int arr1[10] = {12, 13, 25, 34, 45, 54, 55, 61, 70, 83};

int binSearch(int cari){

    int found = 0;
    int i, n, k;
    i = 0;
    n = 9;

    while((i <= n) && (found == 0)){
        k = (int)(i + n) / 2;
        if(arr1[k] == cari){
            found = 1;
        }
        else{
            if(arr1[k] > cari){
                n = k - 1;
            }
            else{
                i = k + 1;
            }
        }
    }

    return found;
}

int main(){

    int cari;
    scanf("%d", &cari);

    if(binSearch(cari) == 1){
        printf("Data ditemukan.\n");
    }
    else{
        printf("Data tidak ditemukan.\n");
    }

    return 0;
}

```

LATIHAN

Buatlah program yang mencari angka terbesar kedua dari suatu kumpulan data menggunakan sequential search. Tidak boleh menggunakan sorting.

Format input:

n, panjang array

data array

Input :

7

89 24 75 11 23 81 102

0

Output :

89

Akhir Kata

Penulis mengucapkan terimakasih yang sebesar-besarnya kepada semua elemen yang telah mendukung berjalannya sesi praktikum pada mata kuliah Algoritma dan Pemrograman I kali ini. Semoga apa yang kita dapatkan kali ini bisa menjadi berkah bagi kita semua.

Daftar Pustaka

Tim Asisten Pemrograman Algoritma dan Pemrograman 2 Angkatan 10. (2019). Array dan Implementasi For dalam Array. Modul Praktikum Algoritma dan Pemrograman 2. Bandung, Jawa Barat, Indonesia