

PRAKTIKUM ANALISIS ALGORITMA

KELAS A



Disusun oleh :

Fidriyanto Rizkillah

(140810160043)

S-1 Teknik Informatika
Fakultas Matematika & Ilmu Pengetahuan Alam
Universitas Padjadjaran
Jalan Raya Bandung - Sumedang Km. 21 Jatinangor 45363

Studi Kasus 1: Pencarian Nilai Maksimal

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

int main()
{
    int n;
    int x[10];
    cout << "Masukkan n : ";
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        cout << "Masukkan Data ke - " << i+1 << " : ";
        cin >> x[i];
    }

    int max = x[0];
    int i = 1;
    while (i <= n)
    {
        if (x[i] > max)
            max = x[i];
        i++;
    }
    cout << "Data terbesar: " << max << endl;

    return 0;
}
```

Studi Kasus 2: Sequential Search

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    int x[10];
    cout << "Masukkan n : ";
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        cout << "Masukkan Data ke - " << i+1 << " : ";
        cin >> x[i];
    }

    int y;
    cout << "Masukkan yang dicari : ";
    cin >> y;

    int i = 0;
    bool found = false;
    int idx;
    while ((i < n) && (!found))
    {
        if (x[i] == y)
            found = true;
        else
            i++;
    }
    if (found)
        idx = i+1;
    else
        idx = 0;

    cout << "Yang dicari berada di urutan : " << idx << endl;
    return 0;
}
```

Studi Kasus 2: Binary Search

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    int x[10];
    cout << "Masukkan n : ";
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        cout << "Masukkan Data ke - " << i+1 << " : ";
        cin >> x[i];
    }

    int y;
    cout << "Masukkan yang dicari : ";
    cin >> y;

    int i = 0;
    int j = n-1;
    bool found = false;
    int idx;
    int mid;
    while ((i <= j) && (!found))
    {
        mid = (i + j)/2;
        if (x[mid] == y)
            found = true;
        else
        {
            if (x[mid] < y)
                i = mid + 1;
            else
                j = mid - 1;
        }
    }
    if (found)
        idx = mid+1;
    else
        idx = 0;
    cout << "Yang dicari berada di urutan : " << idx << endl;

    return 0;
}
```

Studi Kasus 2: Insertion Search

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    int x[10];
    cout << "Masukkan n : ";
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        cout << "Masukkan Data ke - " << i+1 << " : ";
        cin >> x[i];
    }
    cout << "Data Sebelum di Sorting : ";
    for (int i = 0; i < n; i++)
        cout << x[i] << " ";
    cout << endl;

    int insert;
    int j;
    for (int i = 1; i < n; i++)
    {
        insert = x[i];
        j = i-1;
        while ((j >= 0) && (x[j] > insert))
        {
            x[j+1] = x[j];
            j--;
        }
        x[j+1] = insert;
    }

    cout << "Data setelah di Sorting : ";
    for (int i = 0; i < n; i++)
        cout << x[i] << " ";

    return 0;
}
```

Studi Kasus 2: Selection Search

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    int x[10];
    cout << "Masukkan n : ";
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        cout << "Masukkan Data ke - " << i+1 << " : ";
        cin >> x[i];
    }
    cout << "Data Sebelum di Sorting : ";
    for (int i = 0; i < n; i++)
        cout << x[i] << " ";
    cout << endl;

    int imax;
    int temp;
    for (int i = n-1; i >= 1; i--)
    {
        imaks = 0;
        for (int j = 1; j <= i; j++)
        {
            if (x[j] > x[imax])
                imax = j;
        }
        temp = x[i];
        x[i] = x[imax];
        x[imax] = temp;
    }

    cout << "Data setelah di Sorting : ";
    for (int i = 0; i < n; i++)
        cout << x[i] << " ";
}
```