

MODUL 13 PENCARIAN

1. Deskripsi Singkat

Istilah pencarian yang dibahas pada modul ini terbatas pada definisi menemukan suatu data dengan nilai tertentu pada sekumpulan data yang bertipe sama. Pada modul 13 ini akan dibahas tentang pencarian, algoritma pencarian, dan juga penggunaannya dalam program.

2. Tujuan Praktikum

Setelah praktikum pada modul 13 ini diharapkan:

1. mahasiswa memahami dan menguasai algoritma pencarian sekuensial dan pencarian biner.
2. mahasiswa mampu mengimplementasikan pencarian secara sekuensial dan biner menggunakan Bahasa Pascal.

3. Material Praktikum

Kegiatan pada modul 13 ini memerlukan material berupa program Pascal.

4. Kegiatan Praktikum

Pada modul ini kita akan membahas tentang pencarian. Proses pencarian adalah proses menemukan data tertentu di dalam sekumpulan data yang bertipe sama. Pada modul ini, sekumpulan data yang bertipe sama akan kita simpan menggunakan array.

Pencarian adalah salah satu hal yang fundamental dalam pemrograman. Jika kita membuka aplikasi pada komputer, fitur yang hampir pasti selalu ada adalah fitur pencarian. Entah itu mencari sebuah kata dalam kumpulan teks, mencari data tertentu pada spreadsheet, mencari mahasiswa dengan nama tertentu pada aplikasi database dan lain sebagainya.

Pada modul praktikum ini akan dibahas 2 metode pencarian dasar yaitu pencarian sekuensial dan pencarian biner.

A. Pencarian Sekuensial pada Array Tidak Terurut

Pada pencarian sekuensial pada array yang tidak terurut, proses pencarian adalah proses iterasi yang dimulai dari data indeks pertama pada array sampai data yang dicari ditemukan atau sampai data pada indeks terakhir.

Dari pernyataan di atas, terdapat 2 kondisi dimana proses pencarian berhenti yaitu:

1. Jika data yang dicari ditemukan
2. Jika sudah sampai di data indeks terakhir.

Berikut adalah fungsi pencarian sekuensial untuk mencari data integer pada array of integer yang tidak terurut.

```
Type Larik = array [1..100] of integer;
Function SequentialSearch(A:Larik; N:integer; X:integer) :
integer;
Var
    i:integer;
Begin
    i:=1;
    while (i<N) and (A[i] <> X) do i:= i + 1;
    if (A[i] <> X) then SequentialSearch:=0
    else SequentialSearch:=i;
End;
```

dimana i adalah indeks array, N adalah jumlah data, X adalah nilai yang mau dicari.

Hasil dari fungsi di atas adalah nomor indeks array dimana data X ditemukan atau 0 jika tidak ada data yang dicari pada array.

Ketik ulang fungsi di atas pada compiler dan simpan dengan nama **praktikum13A.pas**. Tambahkan bagian main program agar dapat menjalankan Function SequentialSearch. Jalankan praktikum13A.pas dan cobalah mencari beberapa nilai X yang **ada** maupun yang **tidak ada** pada array Anda buat.

B. Pencarian Sekuensial pada Array Terurut

Pada pencarian sekuensial pada array yang terurut, proses pencarian adalah proses iterasi yang dimulai dari data indeks pertama pada array sampai data yang dicari ditemukan atau nilai yang dicari sudah melebihi atau kurang dari nilai data pada indeks array yang dikunjungi atau sampai data pada indeks terakhir.

Dari pernyataan di atas, terdapat 3 kondisi dimana proses pencarian berhenti yaitu:

1. Jika data yang dicari ditemukan
2. Jika nilai data pada indeks array yang sedang dikunjungi sudah melebihi atau kurang dari nilai yang dicari (tergantung array terurut menaik atau menurun).
3. Jika sudah sampai di data indeks terakhir.

Berikut adalah fungsi pencarian sekuensial untuk mencari data integer pada array of integer yang **terurut menaik (dari kecil ke besar)**.

```
Type Larik = array [1..100] of integer;
```

```

Function Sequentialsearch(A:Larik; N:integer; X:integer):
integer;
Var
    i:integer;
Begin
    i:=1;
    while (i<N) and (A[i] < X) do i:= i + 1;
    if (A[i] = X) then Sequentialsearch:=i
    else Sequentialsearch:=0;
End;

```

dimana i adalah indeks array, N adalah jumlah data, X adalah nilai yang mau dicari. Hasil dari fungsi di atas adalah nomor indeks array dimana data X ditemukan atau 0 jika tidak ada data yang dicari pada array.

Ketik ulang fungsi tersebut pada compiler dan lengkapi dengan bagian main program. Simpan dengan nama **praktikum13B_naik.pas**. Jalankan program, cobalah buat sebuah array dengan isi terurut menaik dan cari beberapa nilai X yang **ada** maupun yang **tidak ada** pada array tersebut.

Simpan ulang praktikum13B_naik.pas menjadi **praktikum13B_turun.pas**. Lakukan modifikasi pada `Function Sequentialsearch` untuk mencari data integer pada array of integer yang **terurut menurun (dari besar ke kecil)**.

C. Pencarian Biner

Pencarian biner atau pencarian bagi dua hanya bisa dilakukan pada array yang terurut. Performa dari pencarian biner jauh lebih cepat dibandingkan pencarian sekuensial. Fungsi pencarian biner dapat dinyatakan sebagai fungsi rekursif atau iterative. berikut ini adalah fungsi pencarian biner secara iterative.

```

program Pencarian;
Type Larik = array [1..100] of integer;

Function BinarySearch(A:larik; N: Integer; X:Integer):
integer;
Var
    low, mid, high: integer;

```

```

begin
    low := 1;
    high := N;
    while (low <= high) do
    begin
        mid := (low + high) div 2;
        if(A[mid] > X) then high := mid - 1
        else if(A[mid] < X) then low := mid + 1
        else break;
    end;
    if A[mid] = X then BinarySearch:= mid {ditemukan}
    else BinarySearch := 0; {tidak ditemukan}
end;
var
    A: larik;
    i,n,x : integer;

begin
    n:=10;
    x:=500;
    for i :=1 to n do A[i]:= i*100;
    writeln(BinarySearch(A,n,x));
end.

```

dimana X adalah nilai yang hendak dicari, low adalah indeks array sebelah kiri, high adalah indeks array sebelah kanan, dan mid adalah indeks array tengah.

Ketik ulang program Pencarian di atas dan simpan dengan nama **praktikum13C.pas**. Jalankan program. Pastikan tidak ada error dan program menghasilkan output yang benar. Jika Anda perhatikan, program pencarian biner di atas adalah untuk array dengan isi terurut menaik. Lakukan modifikasi pada Function BinarySearch untuk mencari data integer pada array of integer yang **terurut menurun (dari besar ke kecil)**.

5. Penugasan

- 1) Buatlah program entri data mahasiswa dengan memanfaatkan record dengan struktur nim, nama, kelas, dan nilai_alpro. Program memiliki menu untuk menginput data, menampilkan data yang telah diinputkan, menampilkan nilai terbesar, nilai terkecil, dan rata-rata. Semua menu dibuat menggunakan sub program.

```
Selamat Datang di Program entri mahasiswa
1. Input Data
2. Tampilkan Data
3. Nilai Alpro Terkecil
4. Nilai Alpro Terbesar
5. Rata-rata Nilai Alpro
6. Keluar
```

- 2) Kemudian, tambahkan menu untuk mencari mahasiswa berdasarkan nama untuk kemudian menampilkan nilai alpronya

```
Selamat Datang di Program entri mahasiswa
1. Input Data
2. Tampilkan Data
3. Nilai Alpro Terkecil
4. Nilai Alpro Terbesar
5. Rata-rata Nilai Alpro
6. Cari Nilai Mahasiswa
7. Keluar
Pilihan Anda (1-7):
```

Jika pengguna memilih nomor 6 maka akan muncul permintaan nama mahasiswa:

```
6. Cari Nilai Mahasiswa
Masukkan nama mahasiswa: Panji
Data Ditemukan! Nilai Alpro Panji = 90
```

Jika nama mahasiswa yang diinput tidak ditemukan, maka muncul pesan seperti berikut ini:

```
6. Cari Nilai Mahasiswa
Masukkan nama mahasiswa: Andi
```

Data Tidak Ditemukan!

Menu pencarian boleh dibuat menggunakan pencarian sekuensial ataupun pencarian biner.

- 3) Terakhir, tambahkan menu untuk mengurutkan mahasiswa berdasarkan nama dan juga mengurutkan mahasiswa berdasarkan nilai Alpro

Selamat Datang di Program entri mahasiswa

1. Input Data
2. Tampilkan Data
3. Nilai Alpro Terkecil
4. Nilai Alpro Terbesar
5. Rata-rata Nilai Alpro
6. Cari Nilai Mahasiswa
7. Urutkan Berdasarkan Nama
8. Urutkan Berdasarkan Nilai Alpro
9. Keluar

Pilihan Anda (1-9) :

Jika pengguna memilih nomor 7 maka akan muncul pesan bahwa data telah diurutkan berdasarkan nama:

Pilihan Anda (1-9) : 7

Data Telah Diurutkan Berdasarkan Nama!

Dan jika pengguna memilih nomor 8 maka akan muncul pesan bahwa data telah diurutkan berdasarkan nilai Alpro:

Pilihan Anda (1-9) : 8

Data Telah Diurutkan Berdasarkan Nilai Alpro!

Pengurutan Nama sesuai dengan abjad dari kecil ke besar dan pengurutan nilai dilakukan secara descending dari besar ke kecil.

Untuk melihat apakah data telah terurut dengan benar maka pengguna bisa memilih menu nomor 2 untuk menampilkan data.

Simpan program dengan nama kelas_nim_praktikum13.pas. Unggah file pada Google Classroom sesuai dengan batas waktu yang telah ditetapkan. Keterlambatan pengumpulan dikenakan sanksi pemotongan nilai sebesar 10 poin per jam.