

Worksheet pertemuan 13 - 1
Algoritma dan Struktur Data
Pencarian Biner

NIM: 20523164

Nama: Fajrun Shubhi

A. Membuat Folder Untuk Menyimpan Hasil Praktikum

1. Siapkan folder kosong dengan nama menggunakan NIM masing-masing. Jika folder NIM pada pertemuan sebelumnya mau dimanfaatkan, jangan lupa pindahkan dulu isinya ke folder lain sebagai arsip.
2. Folder ini akan dijadikan tempat untuk menyimpan semua pdf dari worksheet ini beserta fail praktikum lainnya.

B. Mengimplementasikan Pencarian Biner dalam Bahasa Java

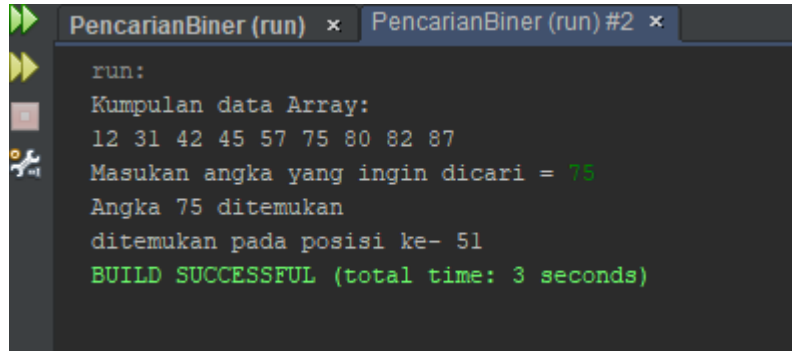
1. Implementasikan algoritma pencarian biner pada aplikasi demo visualisasi pencarian biner (seperti gambar di bawah ini) dalam Bahasa Java menggunakan Netbeans sebagai subfolder dalam folder NIM.

```
ketemu = false
kiri   = 1
kanan  = banyak
while (not(ketemu) and kiri <= kanan)
    tengah = (kiri+kanan)div 2
    if Data[tengah] = X
        ketemu = true
    else
        if X < Data[tengah]
            kanan = tengah - 1
        if X > Data[tengah]
            kiri = tengah + 1
    if (ketemu)
        posisi = tengah
    else
        posisi = 0
```

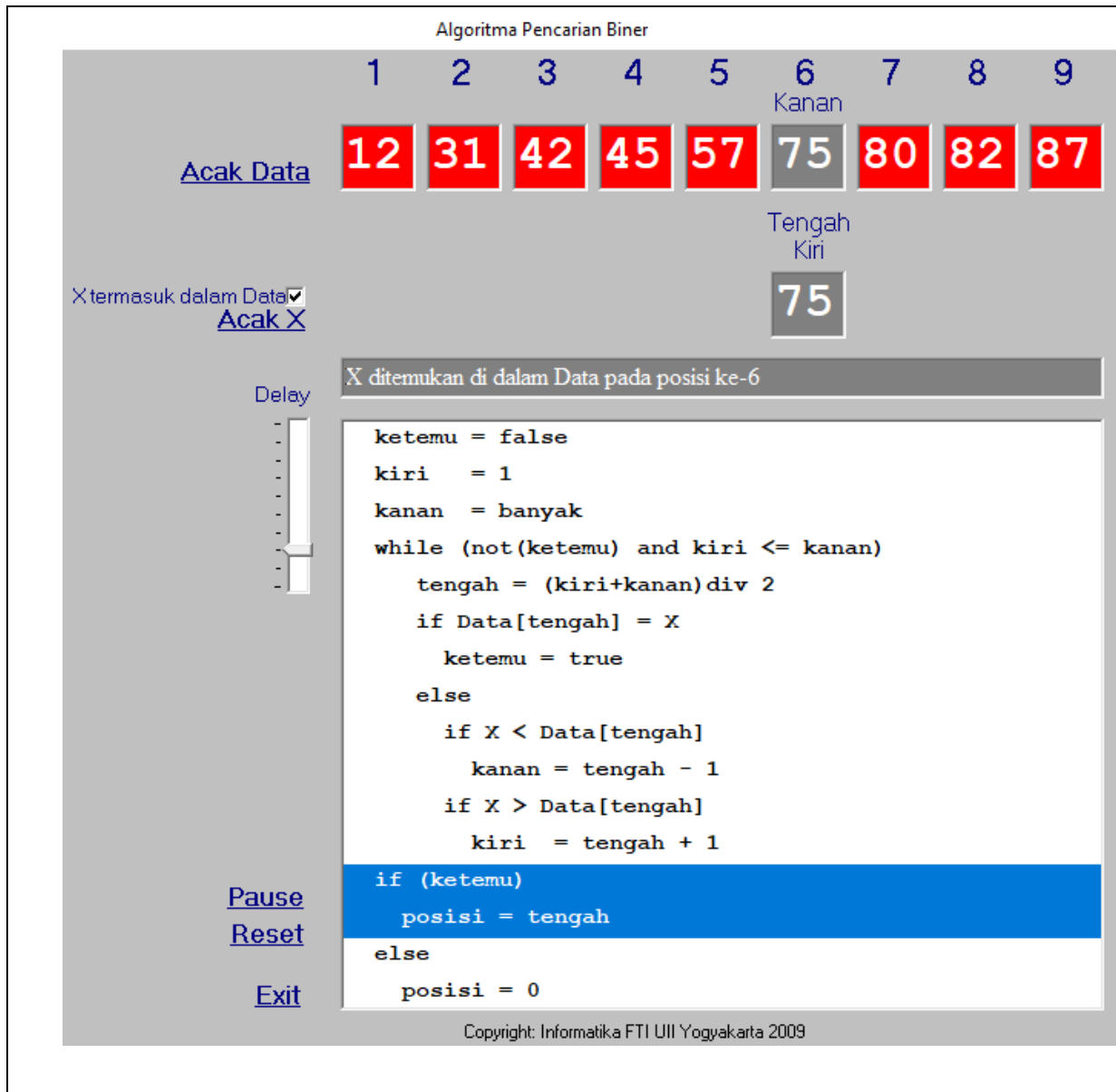
Gambar Algoritma Pencarian Biner

2. Gunakan data sekitar 10 sebagai elemen larik untuk mengujikannya.

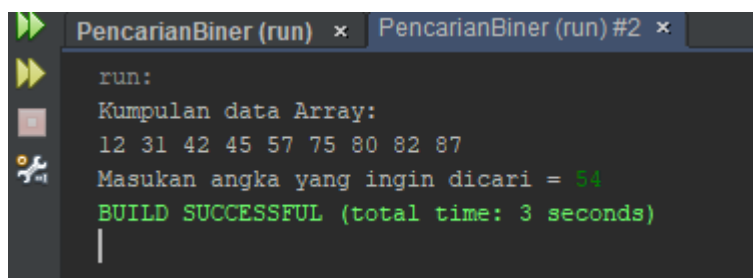
Salin tangkapan layar ketika program dijalankan dengan mensimulasikan proses pencarian X (data yang dicari) sebagai elemen larik Data (X berhasil ditemukan) pada kotak di bawah ini



```
PencarianBiner (run) x  PencarianBiner (run) #2 x
run:
Kumpulan data Array:
12 31 42 45 57 75 80 82 87
Masukan angka yang ingin dicari = 75
Angka 75 ditemukan
ditemukan pada posisi ke- 51
BUILD SUCCESSFUL (total time: 3 seconds)
```



Salin tangkapan layar ketika program dijalankan dengan mensimulasikan proses pencarian X (data yang dicari) yang bukan elemen larik (X tidak berhasil ditemukan) pada kotak di bawah ini



Algoritma Pencarian Biner

	1	2	3	4	5	6	7	8	9
<u>Acak Data</u>	19	21	26	28	35	37	55	59	87

X termasuk dalam Data ☐ Acak X 54

X tidak ditemukan, posisi di-reset ke-0

Delay

```

ketemu = false
kiri   = 1
kanan  = banyak
while (not(ketemu) and kiri <= kanan)
    tengah = (kiri+kanan)div 2
    if Data[tengah] = X
        ketemu = true
    else
        if X < Data[tengah]
            kanan = tengah - 1
        if X > Data[tengah]
            kiri  = tengah + 1
if (ketemu)
    posisi = tengah
else
    posisi = 0
  
```

[Pause](#)
[Reset](#)
[Exit](#)

Copyright: Informatika FTI UII Yogyakarta 2009

***Catatan**

- Jangan lupa simpan juga fail worksheet ini (yang sudah diisi) sebagai fail pdf di folder NIM anda.
- Kompres folder ini sebagai fail ZIP kemudian kumpulkan di classroom atau ruang pengumpulan lain di kelas masing-masing.