

Modul 6 – Searching

1. Buatlah fungsi sequential search (dapat juga menemukan posisi-posisi dari data yang sama), dengan argument atau parameter berupa :

- a. data yang akan dicari
- b. list data dari data yang akan dicari

Sedangkan return value dari fungsi ini berupa:

- a. indeks-indeks atau posisi dari data yang dicari pada list (jika data ditemukan), dan 'Data tidak ada' jika data tidak ditemukan.
- b. Jumlah iterasi yang diperlukan selama proses pencarian

Contoh hasil eksekusi dapat dilihat sebagai berikut :

```
In [2]: a=[1,5,9,8,1,5,10,26,5,12]

In [3]: [hasil,jumlahIterasi]=seqSearch(a,0)
print('Posisi Data=',hasil)
print('Jumlah Iterasi=',jumlahIterasi)

Posisi Data= Data tidak ada
Jumlah Iterasi= 10
```

Pada contoh diatas, data '0' tidak terdapat pada list *a*, dan jumlah iterasi pencarian yang dilakukan sebanyak 10x

```
In [2]: a=[1,5,9,8,1,5,10,26,5,12]

In [4]: [hasil,jumlahIterasi]=seqSearch(a,9)
print('Posisi Data=',hasil)
print('Jumlah Iterasi=',jumlahIterasi)

Posisi Data= [2]
Jumlah Iterasi= 10
```

Pada contoh diatas, data '9' berada di indeks ke-2, dan jumlah iterasi yang dilakukan sebanyak 10x

```
In [2]: a=[1,5,9,8,1,5,10,26,5,12]

In [5]: [hasil,jumlahIterasi]=seqSearch(a,5)
print('Posisi Data=',hasil)
print('Jumlah Iterasi=',jumlahIterasi)

Posisi Data= [1, 5, 8]
Jumlah Iterasi= 10
```

Data '5' berada pada indeks ke-1, 5, dan 8. Dan jumlah iterasi tetaplah 10 kali iterasi

2. Buatlah fungsi ordered sequential search (dapat juga menemukan posisi-posisi dari data yang sama), yaitu pencarian pada list data dimana semua data-data pada list tersebut dalam keadaan terurut (ascending). Argument atau parameter pada fungsi tersebut berupa :

- a. data yang akan dicari
- b. list data dari data yang akan dicari

Sedangkan return value dari fungsi ini berupa:

- a. indeks-indeks atau posisi dari data yang dicari pada list (jika data ditemukan), dan 'Data tidak ada' jika data tidak ditemukan.
- b. Jumlah iterasi yang diperlukan selama proses pencarian

Contoh hasil eksekusi (dengan data pada list *a* sama dengan sebelumnya, hanya saja sudah dalam keadaan terurut) dapat dilihat sebagai berikut :

```
In [7]: a=[1,1,5,5,5,8,9,10,12,26]

In [8]: [hasil,iterasi]=orderedSeqSch(a,0)
print('Posisi data=',hasil)
print('jumlah iterasi=',iterasi)

Posisi data= Data tidak ada
jumlah iterasi= 1
```

Dapat dilihat bahwa data '0' tidak terdapat pada list *a*, dan jumlah iterasi yang dilakukan hanya 1x saja (dibandingkan dengan fungsi sequential search sebelumnya, yang membutuhkan 10x iterasi)

```
In [7]: a=[1,1,5,5,5,8,9,10,12,26]

In [9]: [hasil,iterasi]=orderedSeqSch(a,9)
print('Posisi data=',hasil)
print('jumlah iterasi=',iterasi)

Posisi data= [6]
jumlah iterasi= 8
```

Jika dilakukan pencarian data '9', maka data ditemukan pada indeks ke-6 dan hanya dilakukan 8 x iterasi pada proses pencarian.

```
In [7]: a=[1,1,5,5,5,8,9,10,12,26]

In [10]: [hasil,iterasi]=orderedSeqSch(a,5)
print('Posisi data=',hasil)
print('jumlah iterasi=',iterasi)

Posisi data= [2, 3, 4]
jumlah iterasi= 6
```

Jika dilakukan pencarian data '5' maka akan dihasilkan tiga indeks yaitu, indeks ke-2,3, dan 4. Sedangkan jumlah iterasi yang dilakukan hanyalah sebanyak 6x.

3. Buatlah fungsi binary search yang sudah dimodifikasi, sehingga dapat mencari data yang sama dan mengembalikan indeks-indeks dari data yang sama tersebut. Argument atau parameter pada fungsi tersebut berupa :

- data yang akan dicari
- list data dari data yang akan dicari

Sedangkan return value dari fungsi ini berupa:

- indeks-indeks atau posisi dari data yang dicari pada list (jika data ditemukan), dan 'Data tidak ada' jika data tidak ditemukan.
- Jumlah iterasi yang diperlukan selama proses pencarian

Contoh hasil eksekusi dari fungsi tersebut adalah :

Data yang dicari adalah '5' :

```
In [4]: a=[1,1,5,5,5,8,9,10,12,26]

In [5]: [hasil,iterasi]=binSearch(a,5)
print('Posisi data=',hasil)
print('jumlah iterasi=',iterasi)

Posisi data= [4, 2, 3]
jumlah iterasi= 4
```

Data yang dicari adalah '10' :

```
In [4]: a=[1,1,5,5,5,8,9,10,12,26]

In [6]: [hasil,iterasi]=binSearch(a,10)
print('Posisi data=',hasil)
print('jumlah iterasi=',iterasi)

Posisi data= [7]
jumlah iterasi= 2
```

Data yang dicari adalah '1' :

```
In [4]: a=[1,1,5,5,5,8,9,10,12,26]

In [7]: [hasil,iterasi]=binSearch(a,1)
print('Posisi data=',hasil)
print('jumlah iterasi=',iterasi)

Posisi data= [1, 0]
jumlah iterasi= 3
```

Data yang dicari adalah '20'

```
In [4]: a=[1,1,5,5,5,8,9,10,12,26]

In [8]: [hasil,iterasi]=binSearch(a,20)
print('Posisi data=',hasil)
print('jumlah iterasi=',iterasi)

Posisi data= data tidak ada
jumlah iterasi= 4
```