

# **PYTHON**

## **SOLUSI STUDI KASUS**

lembar solusi ini sebagai salah satu tugas dalam  
mata kuliah Dasar Pemrograman

**dosen pengampu**  
**Indira Syawanodya, M.Kom.**  
**Yulia Retnowati, S.Pd., M.T.**



disusun oleh  
**Andika Eka Kurnia      2306033/1A RPL**

**PROGRAM STUDI REKAYASA PERANGKAT LUNAK**  
**KAMPUS UPI DI CIBIRU**  
**UNIVERSITAS PENDIDIKAN INDONESIA**  
**2023**

File dnsearch.py

```
"""
    Nama   : Andika Eka Kurnia
    NIM    : 2306033
    Kelas  : 1A - RPL
"""
def linear_search(arr, key):
    for i in range(len(arr)):
        if arr[i] == key:
            return i
    return -1

def binary_search(arr, key):
    low = 0
    high = len(arr) - 1

    while low <= high:
        mid = (low + high) // 2
        if arr[mid] == key:
            return mid
        elif arr[mid] > key:
            high = mid - 1
        else:
            low = mid + 1

    return -1
```

## File tugas-1.py

```

"""
    Nama : Andika Eka Kurnia
    NIM : 2306033
    Kelas : 1A - RPL
"""

import dnsearch as dns

barang = ['kunci', 'ember', 'jaket', 'ban', 'mobil', 'sepeda',
          'motor', 'baju', 'celana', 'topi', 'jam', 'sendal', 'sepatu',
          'dompet', 'jaket']

cari_barang = input('Masukkan nama barang yang ingin dicari: ')

hasil_index = dns.linear_search(barang, cari_barang)

if hasil_index != -1:
    print(cari_barang, f"ditemukan di index ke-{hasil_index}")
else:
    print(cari_barang, 'tidak ditemukan')

```

## Terminal tugas-1.py

```

1 """
2     Nama : Andika Eka Kurnia
3     NIM : 2306033
4     Kelas : 1A - RPL
5 """
6
7
8 def linear_search(arr, key):
9     for i in range(len(arr)):
10        if arr[i] == key:
11            return i
12        else:
13            return -1
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

```

PS C:\DikDns\Programming\upi-rl102> python .\pertemuan-11\main.py
[0, 0, 0, 0, 0, 0, 0, 0]
PS C:\DikDns\Programming\upi-rl102> python .\pertemuan-12\tugas-1.py
Masukkan nama barang yang ingin dicari: kunci
kunci ditemukan di index ke- 0
PS C:\DikDns\Programming\upi-rl102> python .\pertemuan-12\tugas-1.py
Masukkan nama barang yang ingin dicari: makanan
jaket tidak ditemukan
PS C:\DikDns\Programming\upi-rl102> python .\pertemuan-12\tugas-1.py
Masukkan nama barang yang ingin dicari: makanan
makanan tidak ditemukan
PS C:\DikDns\Programming\upi-rl102> python .\pertemuan-12\tugas-1.py
Masukkan nama barang yang ingin dicari: kunci
kunci ditemukan di index ke-0
PS C:\DikDns\Programming\upi-rl102>

```

## File tugas-2.py

```

"""
    Nama  : Andika Eka Kurnia
    NIM   : 2306033
    Kelas : 1A - RPL
"""

import dnsearch as dns

nama_mahasiswa = ['Katon', 'Rudi', 'Joko', 'Rina', 'Rudi',
                  'Andika', 'Rita', "Achmad", "Nashirul", "Fatra", "Risti",
                  "Rifiani", "Bagas", "Asep", "Rizal", "Dicky", "Rafi", "Zamzami",
                  "Shandy", "Salsabila"]

cari_mahasiswa = input('Masukkan nama: ')
hasil_index = dns.linear_search(nama_mahasiswa, cari_mahasiswa)

if hasil_index != -1:
    print(cari_mahasiswa, f"ditemukan di index ke-
{hasil_index}")
else:
    print(cari_mahasiswa, 'tidak ditemukan')

```

## Terminal tugas-2.py

```

1 import dnsearch as dns
2
3
4 nama_mahasiswa = ['Katon', 'Rudi', 'Joko', 'Rina', 'Rudi', 'Andika', 'Rita', 'Achmad',
5                  "Nashirul", "Fatra", "Risti", "Rifiani", "Bagas", "Asep", "Rizal",
6                  "Dicky", "Rafi", "Zamzami", "Shandy", "Salsabila"]
7
8 cari_mahasiswa = input('Masukkan nama: ')
9
10 hasil_index = dns.linear_search(nama_mahasiswa, cari_mahasiswa)

```

```

PS C:\DikDns\Programming\upi-rl102> python .\pertemuan-12\tugas-2.py
Masukkan nama: Andika
Andika ditemukan di index ke-5
PS C:\DikDns\Programming\upi-rl102> python .\pertemuan-12\tugas-2.py
Masukkan nama: haha
haha tidak ditemukan
PS C:\DikDns\Programming\upi-rl102> python .\pertemuan-12\tugas-2.py
Masukkan nama: kekeke
kekeke tidak ditemukan
PS C:\DikDns\Programming\upi-rl102> python .\pertemuan-12\tugas-2.py
Masukkan nama: Shandy
Shandy ditemukan di index ke-18
PS C:\DikDns\Programming\upi-rl102>

```

### File tugas-3.py

```
"""
    Nama   : Andika Eka Kurnia
    NIM    : 2306033
    Kelas  : 1A - RPL
"""

from time import perf_counter
import numpy as np
import dnsearch as dns

array = np.array([1, 2, 5, 7, 8, 10, 16, 18, 19, 23, 24, 26, 28,
29, 32, 33, 34, 35, 36, 38, 40, 41, 42, 43, 44, 46, 48, 49, 51,
55, 57, 58, 59, 60, 63, 65, 66, 69, 74, 75, 76, 77, 78, 79, 81,
82, 85, 90, 93, 100])

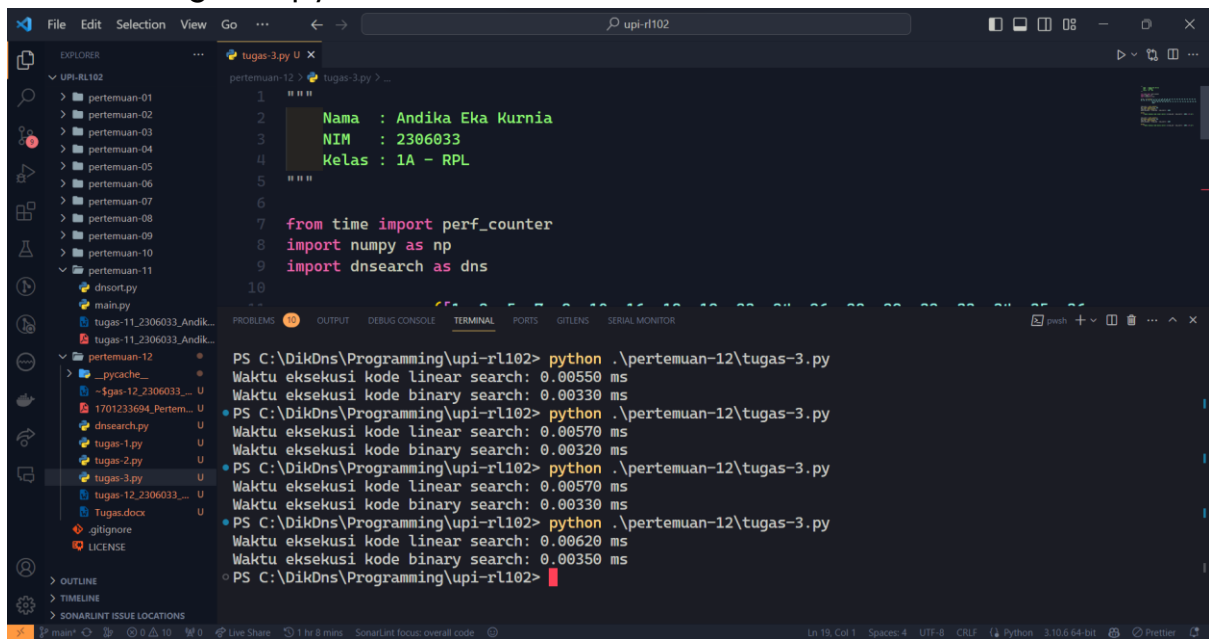
time_start = perf_counter()
dns.linear_search(array, 60)
time_end = perf_counter()

print(f"Waktu eksekusi kode linear search: {(time_end -
time_start) * 1000} ms")

time_start = perf_counter()
dns.binary_search(array, 60)
time_end = perf_counter()

print(f"Waktu eksekusi kode binary search: {(time_end -
time_start) * 1000} ms")
```

## Terminal tugas-3.py



The screenshot shows a Visual Studio Code editor with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The file explorer shows a project named 'UPI-RL102' with several folders and files. The code editor shows a Python script named 'tugas-3.py' with the following content:

```
1 """
2 Nama : Andika Eka Kurnia
3 NIM : 2306033
4 Kelas : 1A - RPL
5 """
6
7 from time import perf_counter
8 import numpy as np
9 import dnsearch as dns
10
```

The terminal shows the execution of the script using the command `python .\pertemuan-12\tugas-3.py`. The output displays the execution time for both linear and binary search algorithms. The results are as follows:

Search Method	Execution Time (ms)
Linear Search	0.00550
Binary Search	0.00330

The terminal output shows that the binary search algorithm is faster than the linear search algorithm for finding the value 60 in the array.

Berdasarkan hasil percobaan, dapat disimpulkan bahwa `binary_search` merupakan algoritma pencarian tercepat untuk mencari nilai 60 dalam array yang sudah terurut tersebut.