

## PERBANDINGAN KECEPATAN DFS DAN BFS

SKENARIO (Banyaknya data)	Kecepatan Pencarian	
	DFS	BFS
25 (data tidak ada)	0.064ms	0.187ms
50 (data ditemukan diposisi paling bawah)	0.167ms	0.107ms
75 (data ditemukan di level 2)	0.036ms	0.049ms

hasil eksperimen menggunakan 25 data

```

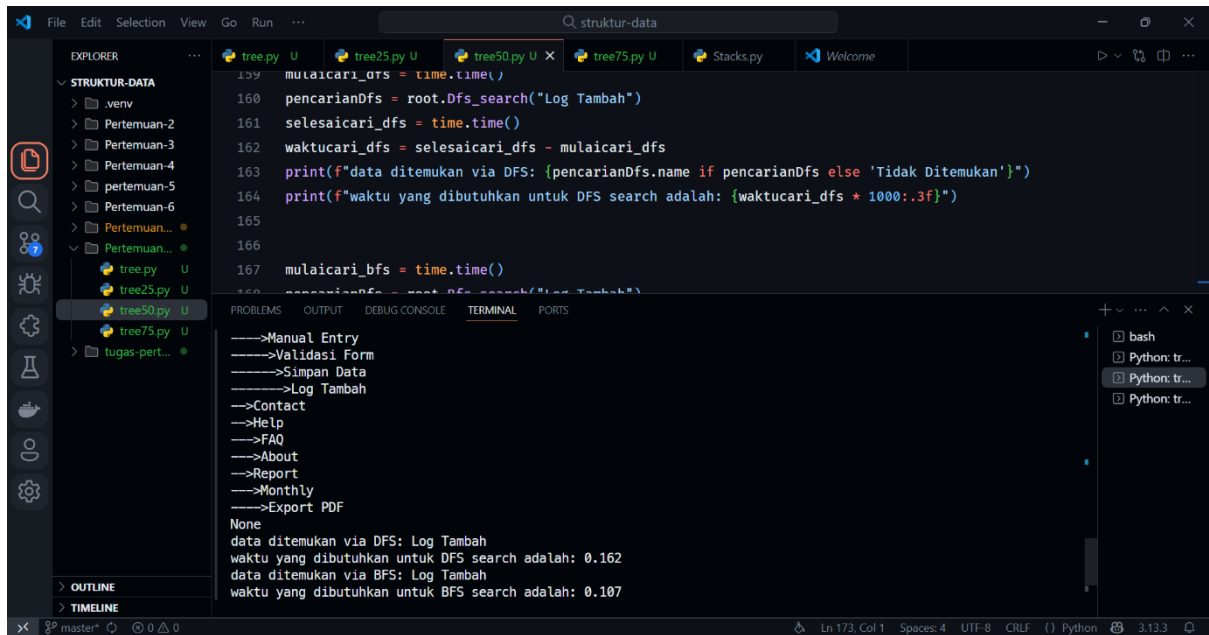
124 pencarianDfs = root.Dfs_search("Tambah dat1a")
125 selesaicari_dfs = time.time()
126 waktucari_dfs = selesaicari_dfs - mulaicari_dfs
127 print(f"data ditemukan via DFS: {pencarianDfs.name if pencarianDfs else 'Tidak Ditemukan'}")
128 print(f"waktu yang dibutuhkan untuk DFS search adalah: {waktucari_dfs * 1000:.3f}")
129
130
131 mulaicari_bfs = time.time()
132 pencarianBfs = root.Bfs_search("Tambah dat1a")
133 selesaicari_bfs = time.time()
134 waktucari_bfs = selesaicari_bfs - mulaicari_bfs
135 print(f"data ditemukan via BFS: {pencarianBfs.name if pencarianBfs else 'Tidak Ditemukan'}")
136 print(f"waktu yang dibutuhkan untuk BFS search adalah: {waktucari_bfs * 1000:.3f}")
137

```

None  
data ditemukan via DFS: Tidak Ditemukan  
waktu yang dibutuhkan untuk DFS search adalah: 0.064  
data ditemukan via BFS: Tidak Ditemukan  
waktu yang dibutuhkan untuk BFS search adalah: 0.187

user@DESKTOP-M806SDK MINGW64 /e/kuliah/struktur-data (master)  
\$

## Hasil eksperimen menggunakan 50 data



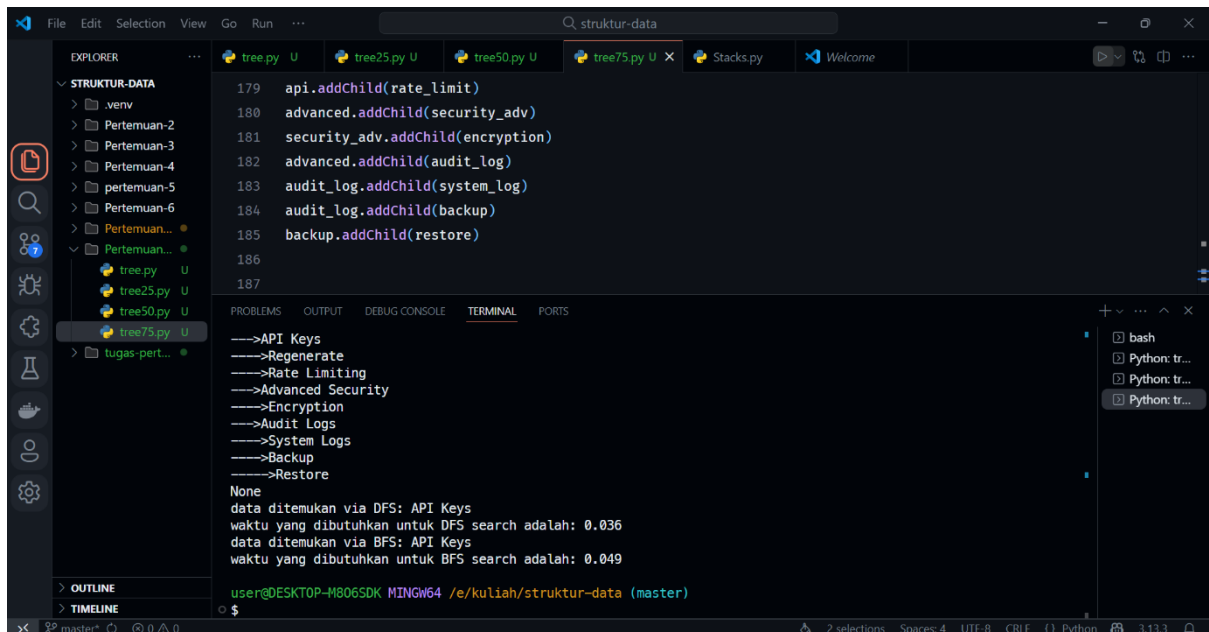
The screenshot shows a VS Code editor with a file explorer on the left containing a directory named 'STRUKTUR-DATA' with subdirectories 'Pertemuan-2' through 'Pertemuan-6'. The main editor displays a Python script 'tree50.py' with the following code:

```
159 mulaicari_dfs = time.time()
160 pencarianDfs = root.Dfs_search("Log Tambah")
161 selesaicari_dfs = time.time()
162 waktucari_dfs = selesaicari_dfs - mulaicari_dfs
163 print(f"data ditemukan via DFS: {pencarianDfs.name if pencarianDfs else 'Tidak Ditemukan'}")
164 print(f"waktu yang dibutuhkan untuk DFS search adalah: {waktucari_dfs * 1000:.3f}")
165
166 mulaicari_bfs = time.time()
167 pencarianBfs = root.Bfs_search("Log Tambah")
```

The terminal output shows the results of the search:

```
---->Manual Entry
---->Validasi Form
---->Simpan Data
---->Log Tambah
---->Contact
---->Help
---->FAQ
---->About
---->Report
---->Monthly
---->Export PDF
None
data ditemukan via DFS: Log Tambah
waktu yang dibutuhkan untuk DFS search adalah: 0.162
data ditemukan via BFS: Log Tambah
waktu yang dibutuhkan untuk BFS search adalah: 0.107
```

## Hasil menggunakan 75 data



The screenshot shows a VS Code editor with a file explorer on the left containing a directory named 'STRUKTUR-DATA' with subdirectories 'Pertemuan-2' through 'Pertemuan-6'. The main editor displays a Python script 'tree75.py' with the following code:

```
179 api.addChild(rate_limit)
180 advanced.addChild(security_adv)
181 security_adv.addChild(encryption)
182 advanced.addChild(audit_log)
183 audit_log.addChild(system_log)
184 audit_log.addChild(backup)
185 backup.addChild(restore)
186
187
```

The terminal output shows the results of the search:

```
---->API Keys
---->Regenerate
---->Rate Limiting
---->Advanced Security
---->Encryption
---->Audit Logs
---->System Logs
---->Backup
---->Restore
None
data ditemukan via DFS: API Keys
waktu yang dibutuhkan untuk DFS search adalah: 0.036
data ditemukan via BFS: API Keys
waktu yang dibutuhkan untuk BFS search adalah: 0.049
```

The terminal prompt shows the user is at the directory `/e/kuliah/struktur-data` (master).

## Kesimpulan

- Berdasarkan hasil eksperimen Ketika menggunakan 25 data dengan pencarian data yang tidak ada maka didapati hasilnya adalah penggunaan algoritma DFS lebih cepat dengan hasil 0.64ms sedangkan menggunakan algoritma BFS didapati hasil 0.187ms.
- Berdasarkan hasil eksperimen Ketika menggunakan 50 data dengan pencarian data yang berada di level terbawah maka didapati hasilnya adalah penggunaan algoritma BFS lebih cepat dengan hasil 0.107ms sedangkan Ketika menggunakan algoritma DFS didapati hasil 0.167ms.
- Berdasarkan hasil eksperimen Ketika menggunakan 75 data dengan pencarian data yang berada di level 2 maka didapati hasilnya adalah penggunaan algoritma DFS lebih cepat dengan hasil 0.36ms sedangkan Ketika menggunakan algoritma BFS didapati hasil 0.49ms.