**LAPORAN**

**PEMROGRAMAN TERSTRUKTUR**

**“ANALISIS ALGORITMA PENGURUTAN”**



Disusun Oleh:

|  |  |
| --- | --- |
| Wildan Mukmin | 2317051080 |

Dosen Pengajar:

Didik Kurniawan M.kom

Tanggal Pengumpulan:

Link Github: https://github.com/WildanMukmin/AnalisisAlgoritmaSorting

Jurusan Ilmu Komputer

Fakultas Matematika dan Ilmu Pengetahuan Alam

Universitas Lampung

2024

# Daftar Isi

[Daftar Isi 2](#_Toc170221794)

[Bab 1: Pendahuluan 3](#_Toc170221795)

[1.1 Latar Belakang 3](#_Toc170221796)

[1.2 Tujuan 3](#_Toc170221797)

[1.3 Algoritma yang Dianalisis 3](#_Toc170221798)

[Bab 2 : Metodologi 4](#_Toc170221799)

[2.1 Lingkungan Pengembangan 4](#_Toc170221800)

[2.2 Implementasi Algoritma 4](#_Toc170221801)

[2.3 Pengujian Kompleksitas 4](#_Toc170221802)

[**Bab 3 : Hasil Pengujian** 5](#_Toc170221803)

[**Bab 4 : Analisis dan Pembahasan** 6](#_Toc170221804)

[**Bab 5: Kesimpulan dan Saran** 7](#_Toc170221805)

[Lampiran 8](#_Toc170221806)

# Bab 1: Pendahuluan

## 1.1 Latar Belakang

## 1.2 Tujuan

## 1.3 Algoritma yang Dianalisis

# Bab 2 : Metodologi

## 2.1 Lingkungan Pengembangan

## 2.2 Implementasi Algoritma

## 2.3 Pengujian Kompleksitas

**Bab 3 : Hasil Pengujian**

**Bab 4 : Analisis dan Pembahasan**

**Bab 5: Kesimpulan dan Saran**

1. Kesimpulan

Evaluasi ekspresi aritmatika menggunakan stack adalah teknik yang bagus dan fleksibel untuk menghitung nilai dari ekspresi aritmatika. Metode ini sangat berguna dalam pemrograman dan aplikasi komputasi lainnya karena kesederhanaan dan efisiensinya. Dengan memahami konsep notasi postfix dan infix serta bagaimana stack digunakan untuk memproses notasi tersebut, kita dapat menangani evaluasi ekspresi aritmatika dengan lebih efektif.

1. Saran

Saran untuk penelitian atau proyek selanjutnya adalah lebih teliti untuk mengevaluasi dan mengolah kode program. Pastikan kita memahami notasi daari postfix. Mengoptimasi explorasi algoritmauntuk mengevaluasi efisiensi kinerja. Dan yang paling penting adalah aktif mengikuti penjelasan dosen dan materi yang diberikan supaya dapat mengerjakan proyek dengan baik,

# Lampiran

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ALGORITMA | KONDISI DATA | N DATA | | | | |
| 10 | 100 | 500 | 1000 | 10000 |
| Bubble Sort | Random Data |  |  |  |  |  |
| Terurut |  |  |  |  |  |
| Terurut Terbalik |  |  |  |  |  |
| Insertion Sort | Random Data |  |  |  |  |  |
| Terurut |  |  |  |  |  |
| Terurut Terbalik |  |  |  |  |  |
| Selection Sort | Random Data |  |  |  |  |  |
| Terurut |  |  |  |  |  |
| Terurut Terbalik |  |  |  |  |  |
| Merge Sort | Random Data |  |  |  |  |  |
| Terurut |  |  |  |  |  |
| Terurut Terbalik |  |  |  |  |  |
| Quick Sort | Random Data |  |  |  |  |  |
| Terurut |  |  |  |  |  |
| Terurut Terbalik |  |  |  |  |  |