Pengurutan Angka Menggunakan Algoritma Pivot Sort dan Radix Sort

http://dx.doi.org/10.28932/jutisi.vXiX.X

Riwayat Artikel

*Received*: 31 Mei 2024 | *Final Revision*: 31 Mei 2024 | *Accepted*: 31 Mei 2024

Creative Commons License 4.0 (CC BY – NC) 

Raden George Samuel Budi🖂 #1

\* Informatika, Universitas Multi Data Palembang  
Jl. Rajawali No.14, Palembang, 30113, Indonesia

1robert.antonius@mhs.mdp.ac.id

🖂Corresponding author: radengeorge@mhs.mdp.ac.id

Abstrak — Penelitian ini mengeksplorasi penerapan algoritma Pivot Sort dan Radix Sort dalam mengurutkan berbagai jenis bilangan, termasuk pecahan, bilangan negatif, bilangan prima, dan bilangan koma. Penelitian bertujuan untuk menyelidiki efisiensi dan efektivitas algoritma pengurutan ini pada rentang data numerik yang beragam. Algoritma Pivot Sort, yang menggunakan elemen pivot untuk membagi dan menaklukkan, dievaluasi bersama algoritma Radix Sort, yang beroperasi dengan mengurutkan digit dari digit paling tidak signifikan hingga digit paling signifikan. Melalui eksperimen dan analisis, penelitian ini bertujuan untuk memberikan wawasan tentang kinerja dan kecocokan algoritma ini untuk mengurutkan dataset numerik yang kompleks, mencakup pecahan, nilai negatif, bilangan prima dan bilangan koma. Hasil dan temuan dari penelitian ini berkontribusi pada pemahaman tentang kemampuan algoritma pengurutan dalam menangani berbagai jenis data numerik dan memberikan rekomendasi untuk pemilihan algoritma yang optimal berdasarkan karakteristik data tertentu dan kebutuhan pengurutan..

Kata kunci— Pivot Sort, Radix Sort, pengurutan angka, pecahan, bilangan negatif, bilangan prima, bilangan koma.

Sorting numbers using the Pivot Sort and Radix Sort algorithms

Abstract — This research explores the implementation of the Pivot Sort and Radix Sort algorithms in sorting various types of numbers, including fractions, negative numbers, prime numbers, and decimal numbers. The study aims to investigate the efficiency and effectiveness of these sorting algorithms across a diverse range of numerical data. The Pivot Sort algorithm, which utilizes a pivot element to divide and conquer, is evaluated alongside the Radix Sort algorithm, which operates by sorting digits from the least significant to the most significant. Through experiments and analysis, this research aims to provide insights into the performance and suitability of these algorithms for sorting complex numerical datasets, encompassing fractions, negative values, prime numbers, and decimal numbers. The results and findings of this research contribute to understanding the capabilities of sorting algorithms in handling various types of numerical data and provide recommendations for optimal algorithm selection based on specific data characteristics and sorting needs.

Keywords— Pivot Sort, Radix Sort, number sorting, fractions, negative numbers, prime numbers, decimal numbers.

1. Pendahuluan

Perkembangan teknologi dunia sangat pesat sekarang. Hal ini terbukti dari kecepatan dan kemudahan dalam penerimaan suatu informasi. Informasi terbentuk dari hasil pemrosesan data. Di dalam buku ReferenceModel for an Open Archival Information System (OAIS) [1]. Algoritma pengurutan, atau yang biasa dikenal sebagai SortingAlgorithm, telah muncul sejak tahun 1956. Algoritma tersebut dikenal dengan nama Sorting by Exchange [2]. Pengurutan data atau Sorting merupakan suatu proses penyusunan kembali beberapa kumpulan data yang sebelumnya telah tersusun secara acak hingga tersusun secara teraturmenurut aturan tertentu. Data ini biasanya bertipe numerik dan karakter [3]. Namun, pada proses pengurutan data acak bilangan integer yang berjumlah besar terutama data bilangan integer dengan banyak data acak lebih dari 1.000 bilangan dapatmenjadi lebih lambat terutama untuk algoritma sorting dengan kompleksitas O(n2) [4]. Algoritma sorting merupakan topik yang penting dalam ilmu komputer dan memiliki berbagai macam algoritma yang berbeda dengan kecepatan dan kompleksitas yang beragam [5]. Pemilihan algoritma sorting yang tepat akan memberikan efisiensi dalam pengolahan data [6] [7].

1. Tujuan

Tujuan dari tugas akhir ini adalah untuk mengurutkan angka yang dari terkecil menjadi terbesar, pengurutan angka ini bisa dilakukan dengan angka-angka yang berbeda, seperti pecahan, min, prima dan bilangan koma, langkah yang dilakukan adalah :

1. Mengurutkan angka menggunakan algoritma pivot sort dan radix sort.
2. Membuat algoritma yang baru untuk melakukan pengurutan angka.
3. Tinjauan Pustaka
4. *Radix Sort*

Radix Sortadalah metode pengurutan yang mengurutkan nilai tanpa melakukan perbandingan pada data yang dimasukkan [8], algoritma ini akan mengurutkan data dengan mengelompokkan data-data berdasarkan digit yang memiliki significant position dan value yang sama. Kelompok digit ini ditampung dalam suatu variable “bucket” [9].

1. Pivot Sort (Quick Sort)

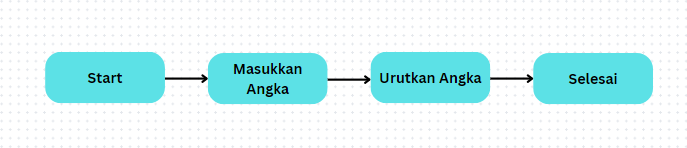
Pivot Sort adalah konsep inti dari Quick Sort yang melibatkan pemilihan pivot, partisi array, dan aplikasi rekursif pada sub-array. Ini adalah algoritma yang sangat efisien dengan kinerja rata-rata O(n log n) dan banyak digunakan dalam pengurutan praktis. Untuk lebih mendalam, Anda dapat merujuk pada sumber-sumber ilmiah yang membahas Quick Sort dalam bahasa Indonesia. Dalam Algoritma Qucik Sort pemilihan pivot sangat menentukan, apakah Quick Sort memberikan kinerja terbaik atau terburuk [10].

1. Metodologi
2. Data Angka

Data yang digunakan akan di input melalui GUI Python, ini tidak menggunakan dataset dikarenaka data yang digunakan akan berbentuk secara langsung. Untuk data angka yang di input akan tergantung pengguna bisa melakukannya secara acak agar bisa melihat seperti apa proses dari mengurutkan angka yang acak menjadi dari yang terkecil ke angka yang terbesar.

1. *Algoritma*

Algoritma yang digunakan adalah Pivot Sort dan Radix Sort. Kedua algoritma ini akan dibandingkan sebuah hasilnya di dalam GUI nya. Angka akan terlihat Ketika melakukan pengurutan angka menggunakan Pivot Sort dan Radix Sort. Dan aka ada juga Tingkat akurasi untuk memastikan seberapa baik Tingkat akurasi dari masing-masing algoritma untuk menentukan pengurutan angka.

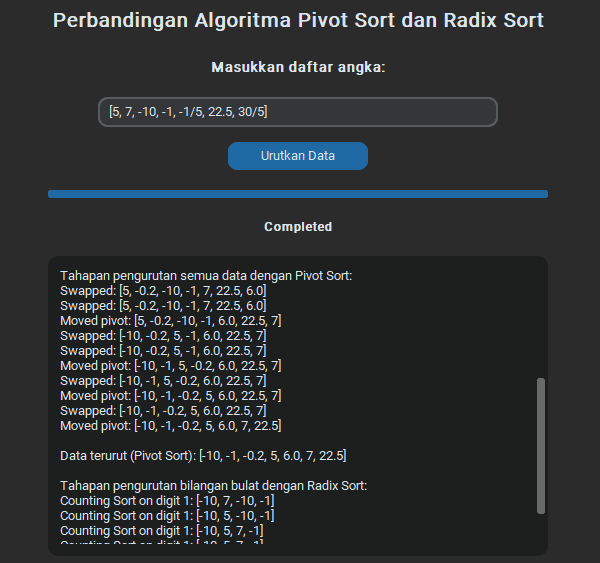


Gambar 1. Flowchart program

1. *Rancangan Program*

Program akan dibuat menggunakan bahasa pemrograman Python dan GUI Python yang digunakan adalah customtkinter :

1. Tampilan GUI.
2. Kolom input angkat untuk melakukan pengurutan
3. Progress Bar yang melihat progress berjalannya tahapan-tahapan yang dikerjakan
4. Text Area yang menjadi tempat untuk melihat hasil-hasil nya.



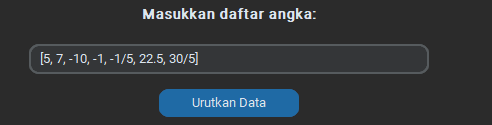
Gambar 2. Rancangan program

1. Implementasi

Implementasi penerapan algoritma yang telah dirancang adalah antara lain melalui:

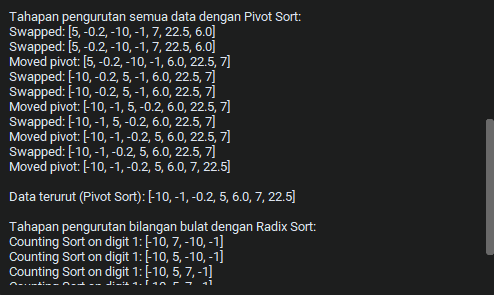
1. Input dan Output Program

Input program berupa angka acak yang bebas bisa dimasukkan oleh pengguna dan tekan urutkan data untuk mengurutkan angka-angka tersebut. Berikut ini cara input angka agar program memiliki output.



Gambar 3. Bentuk input

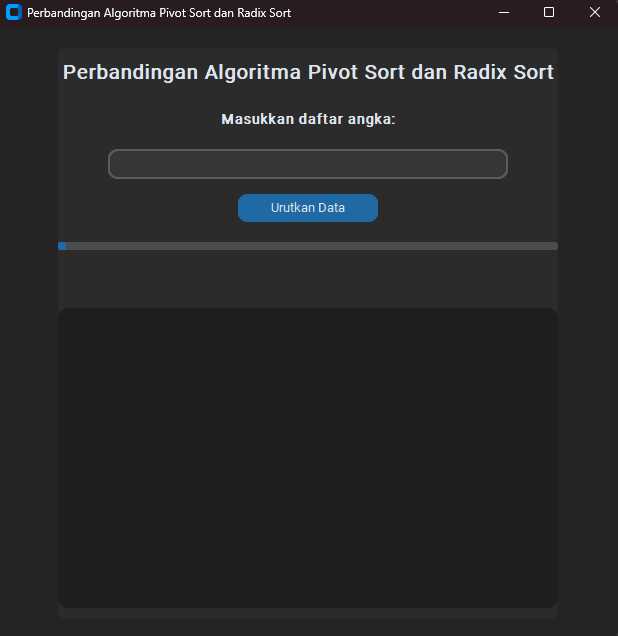
Output program berisi tahapan-tahapan pengurutan angka dari yang terkecil sampai yang terbesar.



Gambar 4. Bentuk output

1. *Tampilan Program (GUI)*

Tampilan GUI menggunakan Python dengan customtkinter.

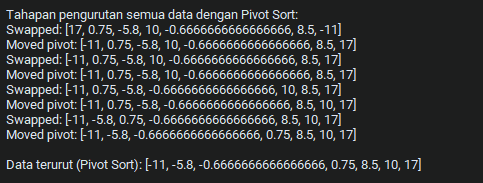


Gambar 5. Tampilan program

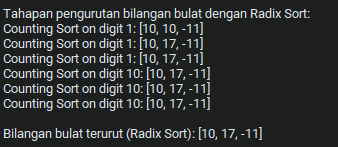
.

1. Pengujian

Pengujian menggunakan angka-angka yang random dengan bilangan prima, pecahan, bilangan koma, dan mines.



Gambar 6. Pengurutan angka dengan Pivot Sort



Gambar 7. Pengurutan angka dengan Radix Sort

1. Kesimpulan dan Saran

Didapat bahwa algoritma telah diimplementasi dengan baik akan tetapi masih mendapatkan ada sebuah kesalahan pada algoritma Radix Sort nya yang tidak melakukan pengurutan angka sesuai dengan input an yang dimasukkan sehingga hasil yang didapatkan tidak begitu memuaskan.

Diharapkan bahwa program ini bisa diperbaiki dan dibuat menjadi lebih baik lagi sehingga tidak terjadi kesalahan yang seperti peneliti buat sekarang ini.

Ucapan Terima Kasih

Terima kasih kepada Universitas Multi Data Palembang yang telah membantu peneliti mempelajari algoritma dan program-program yang memiliki kompleksitas dari terendah sampai tinggi. Peneliti juga ingin mengucapkan terima kasih kepada dosen-dosen yang telah mengajarkan peneliti dengan baik, dan juga mengucapkan terima kasih kepada programmer-programmer terdahulu dan video-video yang berada di sosial media yang telah membantu peneliti juga dan juga kepada ChatGPT terkadang sangat membantu juga disaat-saat yang dibutuhkan atau tidak memiliki jalan keluar lagi. Sekali lagi peneliti ucapkan terima kasih.

Daftar Pustaka

|  |  |
| --- | --- |
| [1] | CCDS, “Reference Model for an Open Archival InformationSystem (OAIS),” in *Reference Model for an Open Archival InformationSystem (OAIS)*, Washington, DC, Magenta Book, 2012, p. 2. |
| [2] | A. Owen, “ Bubble Sort: An Archaeological Algorithmic Analysis,” in *Bubble Sort: An Archaeological Algorithmic Analysis*, Durham, North California, Duke University, 2003, pp. 1-5. |
| [3] | N. G. W. A. S. P. K. Z. I. &. S. A. Sari, “Analisis Algoritma Bubble Sort Secara Ascending Dan Descending Serta Implementasinya Menggunakan Bahasa Pemrograman Java,” *ADI Bisnis Digital Interdisiplin Jurnal,* pp. 16-23, 2022. |
| [4] | D. W. A. P. P. P. &. H. H. Anggreani, “Perbandingan Efisiensi Algoritma Sorting dalam Penggunaan Bandwidth,” *ILKOM Jurnal Ilmiah,* pp. 96-103, 2020. |
| [5] | R. W. &. S. D. Arifin, “Algoritma Metode Pengurutan Bubble Sort dan Quick Sort Dalam Bahasa Pemrograman C++,” *Information System For Educators And Professionals,* pp. 178-187, 2020. |
| [6] | Y. A. R. &. R. A. Puspita Sari, “Perbandingan Efisiensi dengan Algoritma Sorting dalam Penentuan Jarak (Studi Kasus: Pet Shop di Bandar Lampung),” *Perbandingan Efisiensi dengan Algoritma Sorting dalam Penentuan Jarak (Studi Kasus: Pet Shop di Bandar Lampung),* pp. 149-159, 2022. |
| [7] | F. N. Anisya Sonita, “ANALISIS PERBANDINGAN ALGORITMA,” *Jurnal Pseudocode,* vol. II, p. 2, 2015. |
| [8] | E. R. &. F. M. Alfatwa D.F., “Implementasi Algoritma Radix Sort dalam Berbagai Kasus Bilangan Dibandingkan Algoritma Pengurutan yang lain,” *Implementasi Algoritma Radix Sort dalam Berbagai Kasus Bilangan Dibandingkan Algoritma Pengurutan yang lain,* 2015. |
| [9] | A. Rahman, “Analisis Algoritma Radix Sort,” *Analisis Algoritma Radix Sort,* p. 5, 2010. |
| [10] | W. T. Saputro, “Kompleksitas Algoritma Quick Sort,” *Kompleksitas Algoritma Quick Sort,* p. 3, 2018. |