

**ABSTRAK**  
**PERBANDINGAN EFISIENSI ALGORITMA SORTING DALAM**  
**PENCARIAN LOKASI ATM MINI KOTA BANDAR LAMPUNG**

(Studi kasus Pada: ATM Mini Bandar Lampung)

**OLEH:**

**Syifa Annisa**

[annisasyifa995@gmail.com](mailto:annisasyifa995@gmail.com)

ATM (Anjungan Tunai Mandiri) mini yang banyak tersebar di berbagai daerah. Namun permasalahan baru yang muncul dan dapat menghambat perkembangan ATM mini ini adalah tidak diketahuinya lokasi ATM mini terdekat, oleh karena itu, dilakukan penelitian ini untuk mengatasi permasalahan tersebut dengan mengembangkan sistem yang dapat mengetahui lokasi ATM mini terdekat dengan menerapkan perbandingan metode diantaranya *Merge Sort* dan *Shell Sort* sehingga menghasilkan metode yang lebih efisien untuk di gunakan. Penelitian ini menggunakan metode yaitu *Merge sort* adalah metode pengurutan yang menggunakan pola divide and conque, dan *Shell sort* juga dikenal sebagai metode pertambahan menurun. Dan menghasilkan metode pencarian yang paling efisien di gunakan adalah metode shell sort dikarenakan pada proses pengurutan data di array tidak banyak perulangan perbandingan pada array sebelumnya. Selain itu berdasarkan grafik waktu pemrosesan, waktu pemrosesan menggunakan Shell Sort menunjukkan 27Ms, sedangkan waktu pemrosesan menggunakan Merge Sort menunjukkan 45Ms. Lalu Metodologi yang digunakan untuk membangun sistem ini adalah Model Prototype. Model ini merupakan sebuah pendekatan terhadap pengembangan perangkat lunak yang sistematis, dengan beberapa tahapan, yaitu: System Engineering, Analysis, Design, Coding, Testing dan Maintenance. Pada tahapan Testing penelitian ini menggunakan blackbox testing dengan komponen pengujian diantaranya uji *interface*, uji fungsi fitur, uji struktur dan *database*, uji kinerja *loading* dan tingkah laku. Penelitian ini dilakukan untuk membuat sebuah aplikasi untuk menentukan ATM Mini Terdekat berbasis Android, Melakukan perbandingan antara metode *Merge Sort* dan *Shell Sort* dalam penentu ATM Mini terdekat. Pencarian ATM Mini terdekat akan diatur oleh suatu aplikasi berbasis android yang dapat di akses oleh semua orang yang berdomisili di Bandar Lampung.

**Kata Kunci:** ATM Mini, Prototype, Blackbox testing

**ABSTRACT**  
**COMPARISON OF SORTING ALGORITHM EFFICIENCY IN**  
**LOCATION SEARCHING OF MINI ATMS IN BANDAR LAMPUNG**  
**(Case study at: ATM Mini Bandar Lampung)**

**BY:**  
**Syifa Annisa**  
[annisasyifa995@gmail.com](mailto:annisasyifa995@gmail.com)

ATMs (Automated Teller Machines) are distributed in regions. However, a new problem that arises that can delay the development of this mini ATM is that the location of them is unknown. Therefore, this research was carried out to overcome this problem by developing a system that can find the location of the nearest mini ATM by applying a comparison method including Merge Sort and Shell Sorting to produce a method that is more efficient to use. This research used a Merge sort method that uses a divide and conquers pattern and Shell sort as the decreasing addition method. To produce the most efficient search method used is the shell sort method because, in the process of sorting data in an array, there are not many comparisons to the previous array. Furthermore, based on the processing time graph, the processing time using Shell Sort was 27 ms, while the processing time using Merge Sort was 45 ms. The methodology used to build this system was the Prototype Model with several stages, namely: system engineering, analysis, design, coding, testing and maintenance. At the testing stage, this research used the blackbox method to test interface testing, feature function testing, structure and database testing, loading performance testing and behaviour. This research aimed to create an application to determine the Nearest Mini ATM based on Android and to compare the Merge Sort and Shell Sort methods in tagging the nearest Mini ATM. The search for the nearest Mini ATM was managed by an android-based application that can be accessed by people in Bandar Lampung.

**Keywords:** Mini ATM, Prototype, Blackbox testing