ROUTE OPTIMIZATION OF WASTE CARRIER TRUCK USING BREADTH FIRST SEARCH (BFS) ALGORITHM

[1]MUHAMMAD FAHREAL BERNOV, [2]ANI DIJAH RAHAJOE, [3*]BUDI M. MULYO

Department of Informatics Engineering, Universitas Bhayangkara Surabaya

Jl. Ahmad Yani No.114, Ketintang, Kec Gayungan, Surabaya, Jawa Timur, 60231

e-mail: [1]shagi@gmail.com, [2]anidj@gmail.com, [3]budi.mm@ubhara.ac.id

[*]Corresponding author

ABSTRACT

Waste problems have always been the main focus which still occurs in cities and regencies in Indonesia, and also Sidoarjo Regency is no exception. Increasingly rapid population growth is one of the factors in the increasing piles of waste in Sidoarjo Regency. The large number of villages with long distances and a large area means that waste collection cannot be carried out on time, causing accumulation of waste that disrupts residents' daily activities. The waste transportation system in Sidoarjo Regency has so far not been optimal because there are still several problems with the accumulation of waste in several sub-districts.

In this study, an optimal route search system was created using the Breadth First Search and Depth First Search Algorithms as a search comparison in order to make it easier for Sanitation Service officers to carry out the waste transportation process by considering the optimal destination location route according to input from the user. The results of this study will display information on the comparison of the total distance traveled and the total volume traveled by the Breadth First Search and Depth First Search algorithms with different differences.

Keywords: *Breadth First Search, Depth First Search, Route Optimization, Waste Transport Path*

1. INTRODUCTION

Waste is defined as the residual results of activities in solid form, both produced by humans and nature and are considered useless. Garbage is an inseparable part of human life, because in their lifetime, everyone must produce waste. The waste problem has now become a problem that cannot be underestimated because of the impact that can be caused due to poor waste disposal that can affect people's lives in general.

Waste management activities are systematic, comprehensive and continuous activities that include planning, reducing and handling waste. Waste handling activities carried out by the regional government of Sidoarjo Regency are carried out by sorting, collecting, transporting, processing and final processing of waste. Waste collection activities carried out by Sanitation Service officers, starting from transferring waste from household trash to Temporary Shelter (TPS) / Integrated Waste Processing Site (TPST) to Final Processing Site (TPA) while ensuring that waste is separated according to the type of waste.

The process of transporting waste is one of the processes that plays an important role in waste management, especially in the cost sector. The costs involved in waste transportation can be in the form of gasoline to run the waste transportation fleet, and wages for drivers and cleaning workers during active working hours when carrying out waste transportation. The waste transportation system in Sidoarjo Regency has so far not been optimal because there are still several problems with the accumulation of waste in several sub-districts. The causes of this problem are because the locations of TPS in Sidoarjo Regency are scattered and quite far away so that there are delays and the large cost sector in the waste transportation schedule.

2. SUPPORTING THEORY

2.1 Graph

Graph G is defined as a set pair (V, E), written with the notation G = (V,E), in which case V is a non-empty set of vertices or nodes and E is a set of edges or arcs.) that connects a pair of vertices (Munir, 2005).

DOI: https://doi.org/10.54732/jeecs.v7i2.23

In general, a graph can be defined as a collection of points (nodes or vertices) connected by lines (arcs or edges). In short, the graph can be written with the notation G = (V, E) which is:

V : a set of points (nodes or vertices).
: $\{v1, v2, v3, v4, v5, \dots \dots vn\}$
E : a set of lines connecting points to other points (arcs or edges).
: $\{e1, e2, e3, e4, e5, \dots \dots en\}$

2.2 DFS (Depth First Search) Algorithms

The Depth First Search algorithm, or better known as DFS, is a traversal graph algorithm that performs vertex searches with a deep approach. The DFS Search algorithm has priority to visit nodes to the deepest level first. Then if a deadlock is found (no more next nodes), the algorithm will check the previous node that has been visited and is still next to another node that has not been visited and traces that node. In other words, the branch node that is visited first. As an illustration, you can see Figure 1.

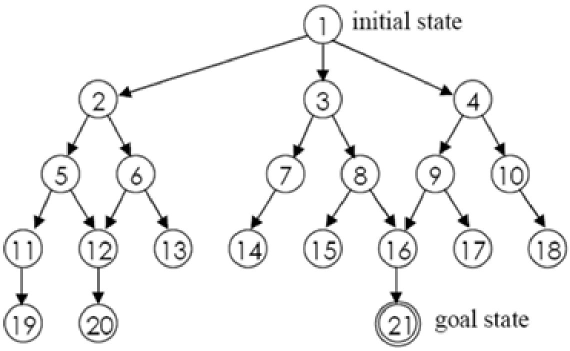


Figure 1. Depth First Search Algorithms

The Depth First Search algorithm has the advantage of being able to quickly reach the depth of the search space. If it is known that the problem solution path will be long then Depth First Search will not waste time doing a large number of shallow states in the graph problem. Depth First Search is much more efficient for search space with many branches because there is no need to execute all nodes at a certain level in the open list. In addition, Depth First Search requires relatively small memory because many nodes on the active path are stored. In addition to the advantages, Depth First Search also has weaknesses including the possibility of not finding the expected destination and only getting one solution for each search.

2.3 BFS (Breadth First Search) Algorithms

Breadth First Search is an algorithm that performs a wide search by visiting a preorder node, namely visiting a node and then visiting all nodes adjacent to that node first. Furthermore, the nodes that have not been visited and are adjacent to the nodes that were visited, and so on. This algorithm requires a queue to store visited nodes. These nodes are needed as a reference to visit adjacent nodes. Each visited node enters the queue only once. This algorithm also requires a boolean table to store visited nodes so that no node is visited more than once.

In the BFS algorithm, the initial node that has been visited is stored in a queue. This queue is used to refer to adjacent nodes that will be visited later in the queue sequence to clarify how the BFS algorithm works and the queue it uses.

Here are the steps for BFS

1. Enter the initial node (root) into the queue.
2. Take a node from the beginning of the queue, then check whether the node is a solution.
3. If the node is a solution, the search is completed and results are returned.
4. If a node is not a solution, queue all adjacent nodes (the initial node) into the queue.
5. If the queue is empty and every node has been checked, the search is complete and returns a solution not found.
6. Repeat the search from step two.

2.3.1. Example of the BFS (Breadth First Search) Algorithm

In the Breadth First Search Algorithm, tracing each point is carried out at a node adjacent to that node (the initial node), as shown in Figure 2 below.

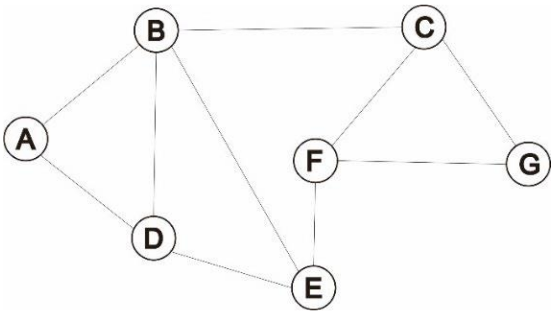


Figure 2. example graph for BFS

For example determining the shortest path with Breadth First Search from A to G, then based on the image above we can implement it into BFS to obtain the shortest path with the following steps.

Step 1

We take the starting point as point A, A has 2 choices of paths, namely towards D and B so that we can get a percentage of the search tree.

Step 2

We search for the shortest path based on the concept of breadth first search as explained earlier, the purpose of the search is to find point E. The search begins by tracing the starting point, namely point A, because point A is the point with the highest level, the search is continued by exploring the points at the level below it or a node adjacent to that node (start node). the next search is to enter the node adjacent to D, namely E. then proceed by entering the vertex that is adjacent to B, namely E and C. The next is to enter the node adjacent to E, namely F.

Then enter the node adjacent to C, namely F and G. In this diagram, point G is the destination point, so the search process stops because it has found a solution, as shown in Figure 3.

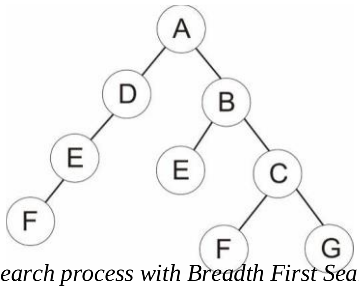


Figure 3. Search process with Breadth First Search method

Based on the above search using the Breadth First Search method, it is found that the shortest path from A to G is A, B, C, G.

3. SYSTEM ANALYSIS AND DESIGN

3.1 System Analysis

Basically this system was developed as a system that implements the Breadth First Search (BFS) algorithm to search for optimal routes in transporting waste around the Sidoarjo district area, which as a transportation method is the Stationary Container System (SCS) or using a Dump Truck as a vehicle reference. This system will process route search data from the starting point to various TPS points specified in the transport schedule obtained from DLHK Sidoarjo Regency.

The data taken is in the form of fleet distribution data and drivers on duty as well as a schedule of garbage collection points for each fleet obtained from the 2021 DLHK Dump Truck Garbage Fleet Service report. If the data has been obtained, then a Graph of the optimal route locations for TPS is made in each sub-district.

Making a Graph is done to find out the route path between TPS location points. for the Graph creation process, researchers will use the Qgis Desktop application to create Graph models at several locations in the three Sidoarjo districts. After making the Graph, the results will be displayed in the system. Following are some of the location data in Table 1, Graph data in Table 2 and Figure 4 Graph models in the Qgis application displayed from several TPS locations in the Buduran sub-district, Sidoarjo city.

Table 1. Table of Buduran District Location Data

Id	Name	Sub-District	Volume	Latitude	Longitude
1	DLHK Sidoarjo City	Buduran	5	-7.432484	112.727690
2	TPS Wadungasih Beseri	Buduran	4	-7.419945	112.7316
3	Buduran Village Depot Rw.02 Buduran Sidoarjo	Buduran	5	-7.428623	112.723114
4	Kupang Sidoarjo Depot Jl. Raya Tebel Sruni Gedangan Sidoarjo	Buduran	5	-7.4095856	112.7270188
5	TPS 3R Sidokerto	Buduran	5	-7.427615	112.711109
7	TPS 3R Sidomulyo	Buduran	4	-7.424991	112.7316
8	TPS 3R Prasung	Buduran	3	-7.422076	112.748749

Tabel 2. Data Graph Table of TPS Buduran District Locations

Id	Id_Origin	Latitude_1	Longitude_1	Id_Destination	Latitude_2	Longitude_2	Range
1	1	-7.432484	112.727690	3	-7.428623	112.723114	0.666
2	1	-7.432484	112.727690	7	-7.424991	112.7316	0.941
3	3	-7.428623	112.723114	5	-7.427615	112.711109	1.341
4	3	-7.428623	112.723114	9	-7.427040	112.723322	0.178
5	7	-7.42499	112.7316	9	-7.427040	112.723322	0.949
6	7	-7.42499	112.7316	2	-7.419945	112.7316	0.562
8	5	-7.42762	112.711109	9	-7.427040	112.723322	1.361
9	9	-7.427040	112.723322	6	-7.412741	112.725288	1.607

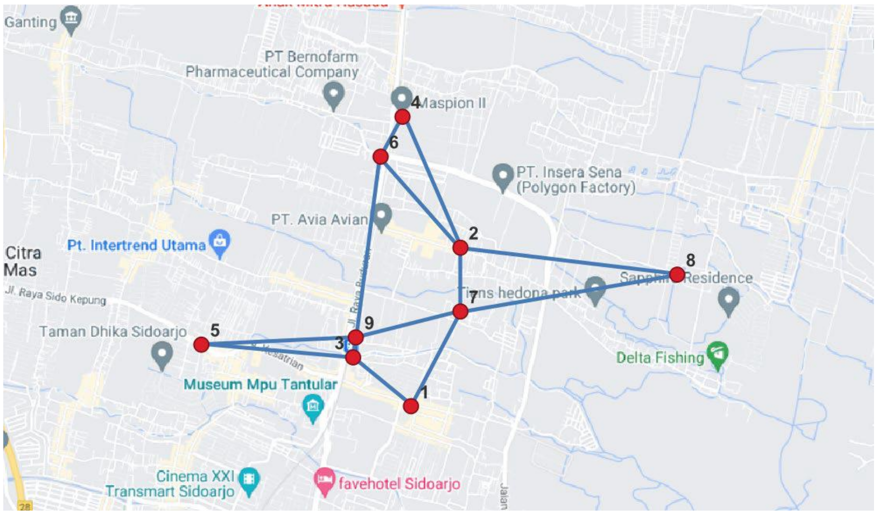


Figure 4. Model graph Location of TPS Buduran District

3.2 System Planning

System planning aims to provide an overview of system planning to be built or developed. This stage will represent how the system flow works both in terms of the user interface or flowchart.

a. Flowchart

Flowcharts are used to analyze, draw or design, document and manage a program. The following is a system flowchart that will be made.

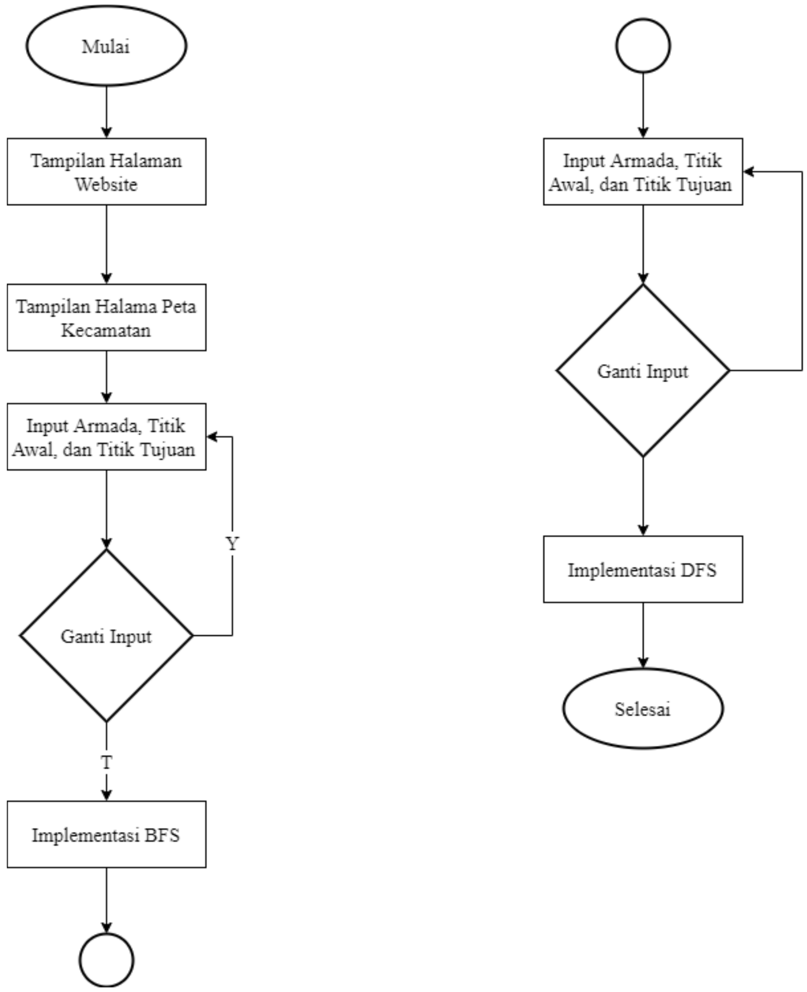

Mulai
Tampilan Halaman Website
Tampilan Halama Peta Kecamatan
Input Armada, Titik Awal, dan Titik Tujuan
Y
Ganti Input
T
Implementasi BFS
Input Armada, Titik Awal, dan Titik Tujuan
Ganti Input
Implementasi DFS
Selesai

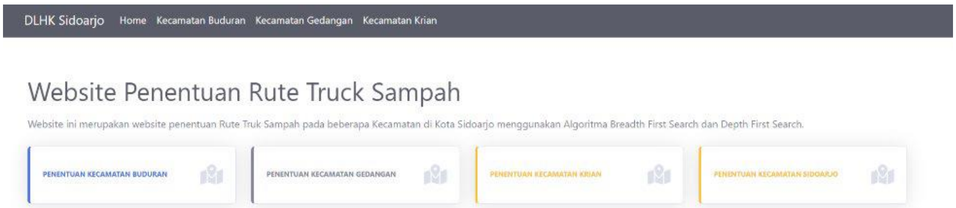
Figure 5. Program flowcharts

Flowchart explanation is as follows:

Through the website page, the user will choose from the three sub-districts where the route determination process is carried out. After that, it will be redirected to the selected district page. To carry out the BFS Algorithm process, the user will input the fleet number, starting location, and destination location. After that the User is given a choice whether the input is correct or not, if it is correct then the System will process the implementation of the BFS Algorithm and the output results will be displayed. Continuing to process the DFS Algorithm, the user will repeat the same sequence, namely inputting the fleet number, initial location, and destination location, then the system will process the implementation of the DFS Algorithm with the displayed output results.

b. Interface design

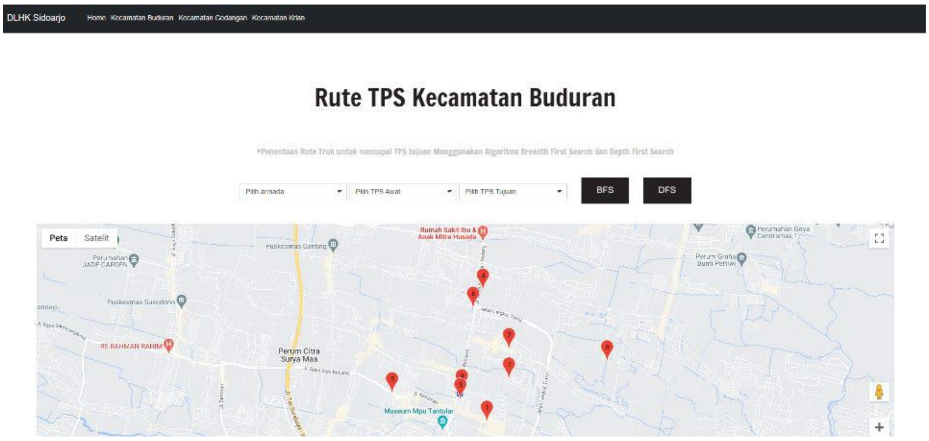
The interface is a form of graphic display that is directly related to the user (User). The user interface serves as a means of connecting between the user and the operating system. In implementing this interface, it will be explained how the truck route optimization system interface will be displayed to the user.


DLHK Sidoarjo
Home
Kecamatan Buduran
Kecamatan Gedangan
Kecamatan Krian
PENENTUAN KECAMATAN BUDURAN
PENENTUAN KECAMATAN GEDANGAN
PENENTUAN KECAMATAN KRIAN
PENENTUAN KECAMATAN SIDOARJO

Website Penentuan Rute Truck Sampah

Website ini merupakan website penentuan Rute Truk Sampah pada beberapa Kecamatan di Kota Sidoarjo menggunakan Algoritma Breadth First Search dan Depth First Search.

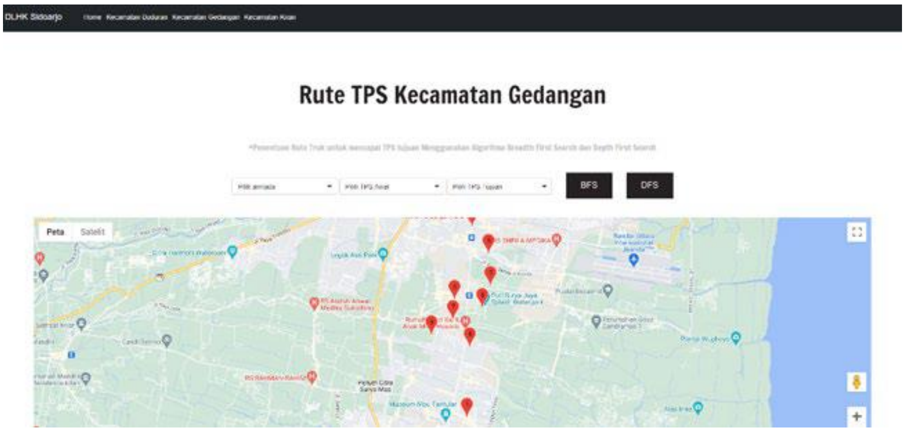
Figure 6. Website Page



Rute TPS Kecamatan Buduran

*Penentuan Rute Truk untuk mencapai TPS tujuan Menggunakan Algoritma Breadth First Search dan Depth First Search

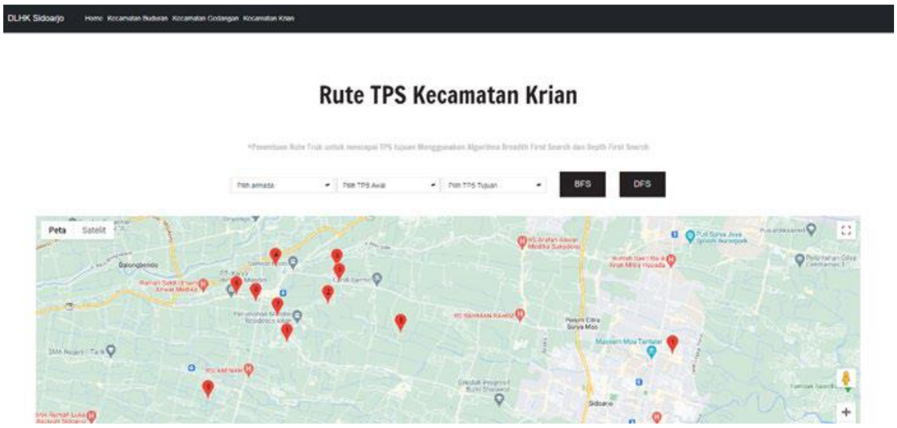
Figure 7. Buduran Sub-District Web Page



Rute TPS Kecamatan Gedangan

*Penentuan Rute Truk untuk mencapai TPS tujuan Menggunakan Algoritma Breadth First Search dan Depth First Search

Figure 8. Gedangan Sub-District Web Page



Rute TPS Kecamatan Krian

*Penentuan Rute Truk untuk mencapai TPS tujuan Menggunakan Algoritma Breadth First Search dan Depth First Search

Figure 9. Krian Sub-District Web Page

4. TESTING AND RESULT

System testing is an advanced stage after system implementation. System testing aims to prove that the system built has been running well in finding the optimal route.

4.1 System Testing

In this test, the system results of the Breadth First Search Algorithm will be compared with the system results of the Depth First Search Algorithm. In this test, the data taken were all TPS locations in 2 sub-districts, namely Buduran and Gedangan sub-districts with a fixed starting point at the Sidoarjo DLHK office. This test is carried out to compare the route passed by the Algorithm (BFS) with the route passed by the Algorithm (DFS) on the system that has been made with the results of the comparison, namely the total distance and total volume of TPS passed.

a. Testing of Buduran Sub-District

The following is a table of the results of a comparison of the total distance and volume by the BFS and DFS algorithms with the route passed in the Buduran sub-district.

Table 4. Total Distance and Total Volume of the BFS Algorithm for Buduran Sub-District

id_origin	Id_destination	*Breadth First Search*	
		Total Distance (Km)	Total Volume m^3
1	2	3.7	7
1	3	1.2	4
1	4	5.5	13
1	5	3.7	9
1	6	4.7	12
1	7	2.1	4
1	8	6.2	7
1	9	1.4	9

Table 5. Total Distance and Total Volume of the DFS Algorithm for Buduran Sub-District

id_origin	id_destination	*Depth First Search*	
		Total Distance (Km)	Total Volume m^3
1	2	3.7	7
1	3	1.2	4
1	4	5.5	13
1	5	6.4	14
1	6	5.2	10
1	7	2.1	4
1	8	6.2	7
1	9	3.8	9

Based on the two tables above, a graph is obtained showing the comparison of the total distance and volume between the BFS and DFS algorithms.

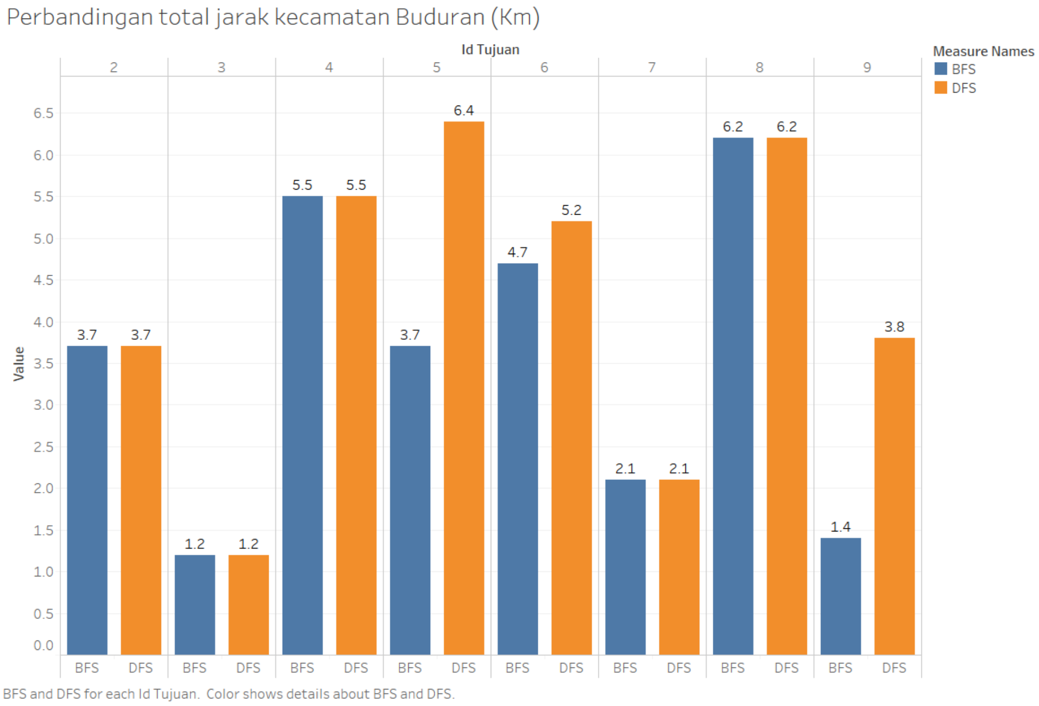


Figure 10. Comparison of the total distance at the point of destination Buduran Sub-District

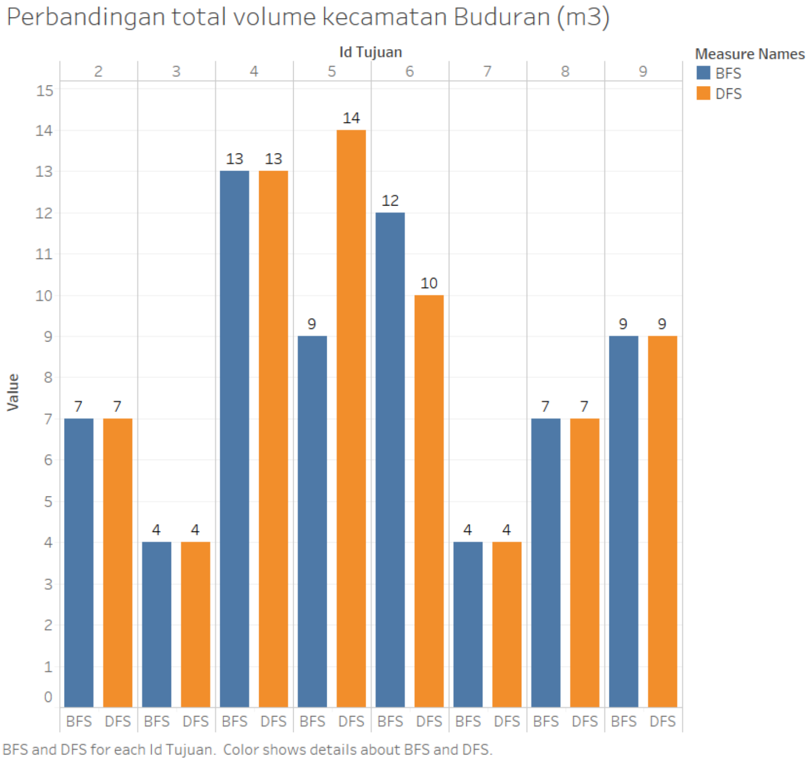


Figure 11. Comparison of the total volume at the point of destination Buduran Sub-District

b. Testing of Gedangan Sub-District

The following is a table of the results of a comparison of the total distance and volume by the BFS and DFS algorithms with the route passed in the Gedangan sub-district.

Table 6. Total Distance and Total Volume of the BFS Algorithm for Gedangan Sub-District

id_origin	id_destination	*Breadth First Search*	
		Total Distance (Km)	Total Volume m^3
1	2	9.3	14
1	3	13.7	22
1	4	13.5	21
1	5	7.7	11
1	6	9.7	17
1	7	6.9	8
1	8	5.3	5
1	9	6	3

Table 7. Total Distance and Total Volume of the DFS Algorithm for Gedangan Sub-District

id_origin	id_destination	*Depth First Search*	
		Total Distance (Km)	Total Volume m^3
1	2	14.4	15
1	3	16.8	17
1	4	16.8	16
1	5	14.8	18
1	6	12.8	12
1	7	9.9	6
1	8	5.3	5
1	9	6	3

Based on the two tables above, a graph is obtained showing the comparison of the total distance and volume between the BFS and DFS algorithms.

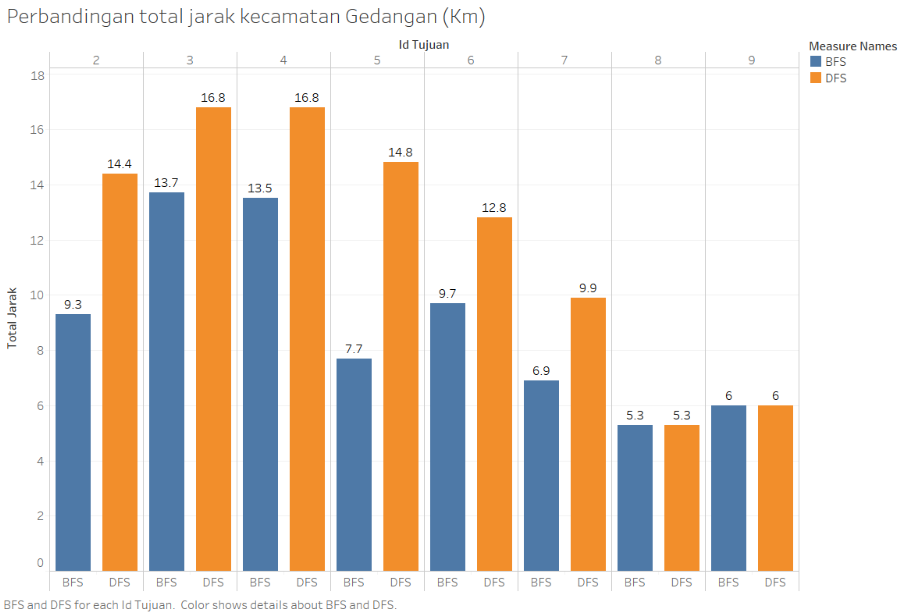


Figure 12. Comparison of the total distance at the point of destination Gedangan Sub-District

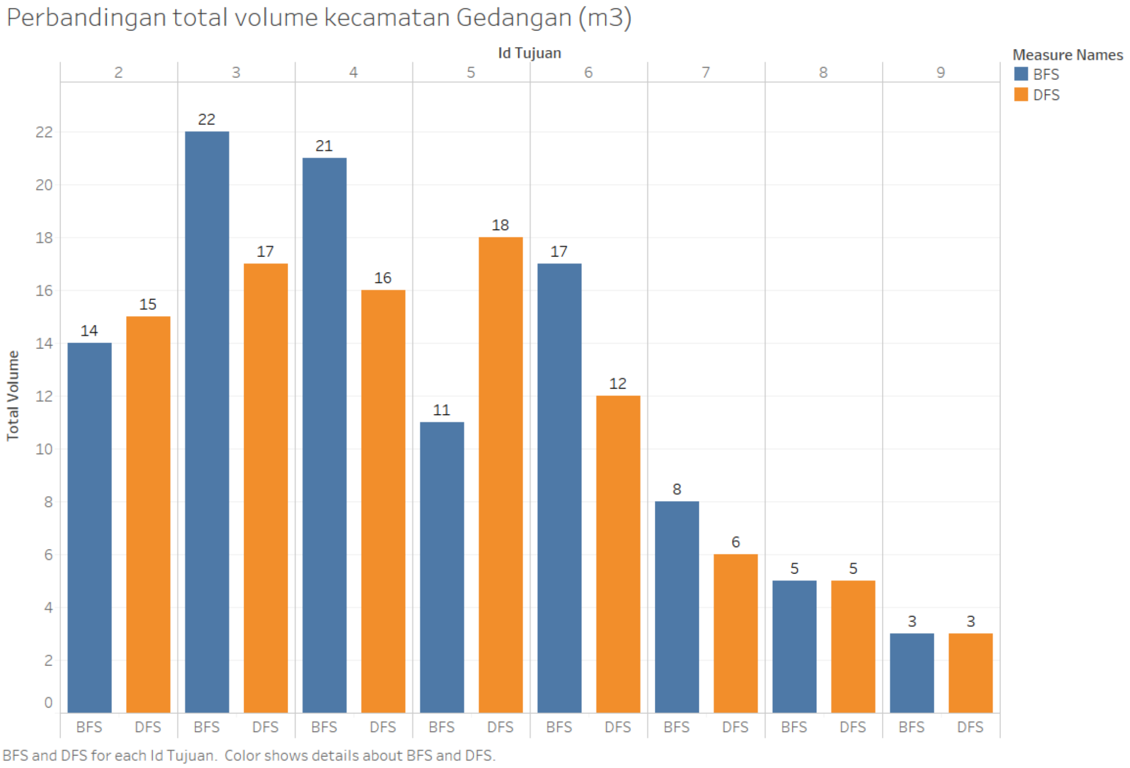


Figure 13. Comparison of the total volume at the point of destination Gedangan Sub-District

4.2 System Testing Analysis

After testing the results of Breadth First Search and Depth First Search in the system can run properly, further tests are carried out by testing the comparison of the processing time of the two algorithms to find out the running time of the processes running in the system. each point of the destination location in the search for the results of the processing time varies because the route search of the two algorithms uses Graph data to process the path traversed to get to the destination.

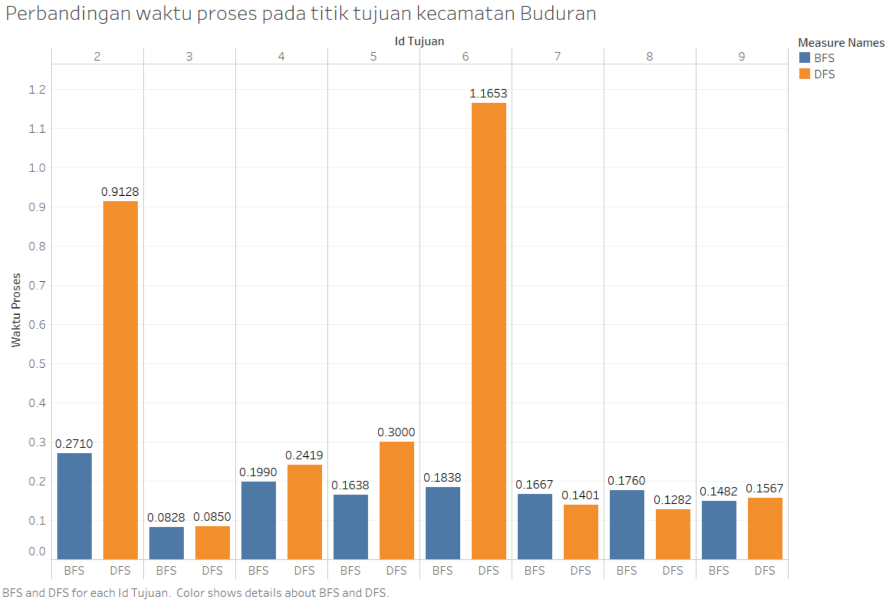


Figure 14. Comparison of Buduran sub-district processing time

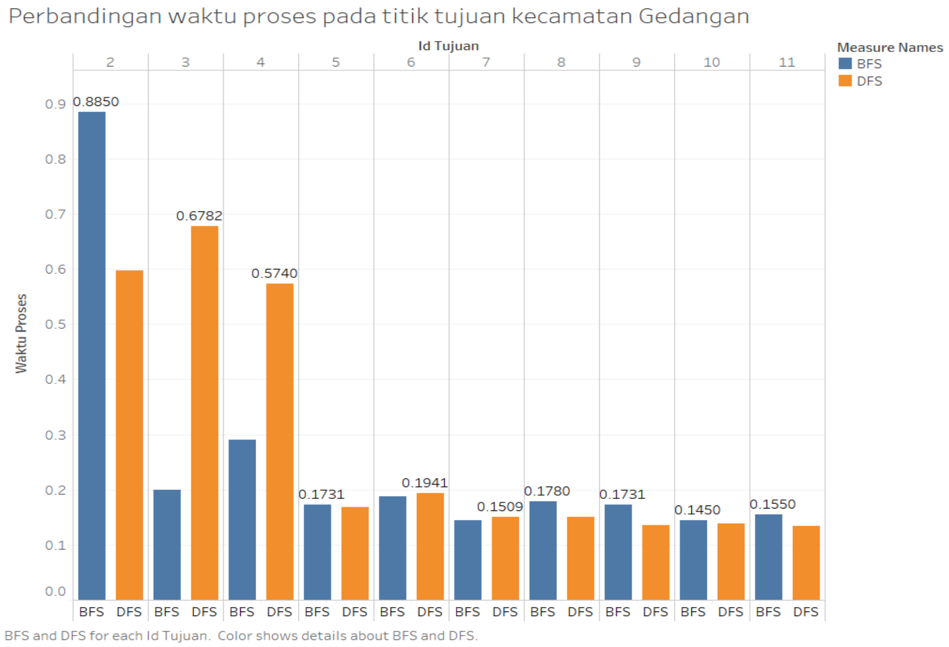


Figure 15. Comparison of Gedangan sub-district processing time

5. CONCLUSIONS AND RECOMMENDATIONS

3.1 Conclusions

The conclusions that can be drawn from the results of this study include the following:

1. This garbage truck route search system can show the optimal destination location point route inputted by the user using the Breadth First Search (BFS) and Depth First Search (DFS) algorithms.
2. Based on the existing test results, the comparison results are obtained with Breadth First Search as the optimal algorithm in the process of mileage and processing time, while depth first search is the optimal algorithm in minimizing the total volume traveled.

5.2 Recommendations

As for suggestions for further research to improve this research, there are several suggestions for developing a better application system, including:

1. Research needs to be done for the shortest path with other algorithms such as Best First Search or others to compare different searches.
2. In this research, web-based applications are expected to be further developed into mobile or desktop-based applications.

REFRENCES

[1] Ari Sellyana, Nur Budi Nugraha (2021), Aplikasi Pendistribusian Barang J&T Dumai Berbasis Mobile Menggunakan Algoritma Breadth First Search. diakses pada 14 Juni 2021.

[2] Surya Tarmiandi, Erna Zuni Astuti, Setia Astuti (2018), Implementasi Algoritma Breadth First Search Pada Pencarian Rute Terpendek Tempat Kos Di Semarang Tengah. diakses pada 14 juli 2021.

[3] Delima Zai, Haeni Budiati, Sunneng Sandino Berutu (2016), Simulasi Rute Terpendek lokasi Pariwisata Di Nias dengan metode Breadth First Search Dan tabu search. diakses pada 14 juli 2021.

[4] Gisella Fransca Laurency Ginting (2019), Perbandingan Algoritma Breadth First Search Dan Steepest Ascent Hill Climbing Dalam Pencarian Rute Kunjungan Rumah Sakit. dikses pada 17 juli 2021.

[5] Anwari, Hozairi (2019), Perbandingan Algoritma Breadth First Search dan Dijkstra Untuk Penentuan Rute Terpendek pengiriman Barang Unilever. Diakses pada 14 juli 2021.

[6] Dwi Budi Marwanto, Arif Arizal, M. Mahaputra Hidayat (2019), Sistem Informasi Geografis Pemetaan Lokasi Perumahan Menggunakan Algoritma Breadth First Search Berbasis Web (Studi Kasus: Kabupaten Sidoarjo). diakses pada 23 april 2022