

BAB II LANDASAN TEORI

2.1 Pengertian Oleh-Oleh

Dari arti harfiahnya, oleh-oleh merupakan barang yang diberikan seseorang dari bepergian, bertujuan untuk dikenang. Oleh-oleh dapat berupa kerajinan dan makanan khas daerah setempat. Oleh-oleh adalah sesuatu yang dibawa dari bepergian, buah tangan. Oleh-oleh berupa kerajinan dalam bahasa Inggris adalah souvenir merupakan kata yang berasal dari bahasa Perancis dengan tulisan yang sama yaitu souvenir, yang berarti “untuk diingat”. Souvenir adalah barang-barang kerajinan tangan (handy crafts), yang merupakan hasil kreativitas para pengrajin yang mampu merubah benda-benda yang terbuang dan tidak berharga menjadi produk kerajinan tangan yang menarik dan diminati banyak orang, terutama para wisatawan.

Dalam kamus *The Collins Cobuild Dictionary* (2009), kata souvenir diartikan:”Souvenir is usually small and relatively inexpensive article given, kept or purchased as a reminder of a place visited, an occasion, etc.” Souvenir adalah benda yang ukurannya relative kecil dan harganya tidak mahal; untuk dihadiahkan, disimpan atau dibeli sebagai kenang-kenangan kepada suatu tempat yang dikunjungi, suatu kejadian tertentu, dsb. Sehingga secara umum oleh-oleh merupakan sesuatu yang dapat dijadikan kenang-kenangan ketika wisatawan kembali ke negara atau daerah asal wisata.

2.2 Metode Floyd Warshall

Algoritma Floyd Warshall adalah salah satu varian dari pemrograman dinamis, metode untuk memecahkan masalah pencarian rute terpendek (sama seperti Algoritma Floyd Warshall). Metode ini melakukan pemecahan masalah dengan memandang solusi yang akan diperoleh sebagai suatu keputusan yang saling terkait. Maksudnya solusi-solusi dibentuk dari solusi yang berasal dari tahap sebelumnya dan ada

kemungkinan solusi lebih dari satu. Algoritma Floyd Warshall merupakan algoritma yang mengambil jarak minimal dari suatu titik ke titik lainnya. Pada algoritma ini menerapkan suatu algoritma dinamis yang menyebabkan akan mengambil jarak lintasan terpendek secara benar. (Siang 2009)

Algoritma Floyd Warshall ini juga bisa diterapkan pada sebuah aplikasi pencari rute jalan yang terdekat dari suatu daerah ke daerah lainnya dengan metode ini hasil yang didapat bisa lebih optimal namun memerlukan resource yang cukup besar jika dipakai untuk mencari kompleks.

Algoritma Floyd Warshall memiliki input graf berarah dan berbobot (V,E) , yang berupa daftar titik (node/vertex V) dan daftar sisi (edge E). Jumlah bobot sisi-sisi pada sebuah jalur adalah bobot jalur tersebut. Sisi pada E diperbolehkan memiliki bobot negatif, akan tetapi tidak diperbolehkan bagi graf ini untuk memiliki siklus dengan bobot negatif. Algoritma ini menghitung bobot terkecil dari semua jalur yang menghubungkan sebuah pasangan titik, dan melakukannya sekaligus untuk semua pasangan titik.

Dalam usaha mencari jalur terpendek, algoritma Warshall memulai iterasi dari titik awalnya kemudian memperpanjang path dengan mengevaluasi titik demi titik hingga mencapai titik tujuan dengan jumlah bobot yang seminimum mungkin. Berikut adalah rumus dasar algoritma Floyd-Warshall:

1. Inisialisasi matriks jarak awal dengan jarak antara node awal dan node tujuan yang dapat langsung dijangkau, dan jarak ke node lainnya diatur menjadi tak terhingga.
2. Untuk setiap pasangan node (i,j) , cari apakah ada jalan yang lebih pendek melalui node tengah k . Jika ada, update jarak antara node i dan j menjadi jarak melalui node k .
3. Ulangi langkah kedua untuk setiap node k yang mungkin menjadi node tengah, hingga semua node telah dihitung.

Setelah selesai, matriks jarak akan berisi jarak terpendek antara setiap pasangan node. Berikut adalah pseudocode algoritma Floyd-Warshall:

```

for k from 1 to |V|
  for i from 1 to |V|
    for j from 1 to |V|
      if dist[i][j] > dist[i][k] + dist[k][j]
        dist[i][j] = dist[i][k] + dist[k][j]

```

Keterangan:

$|V|$ adalah jumlah node pada grafik

$dist[i][j]$ adalah jarak terpendek antara node i dan j

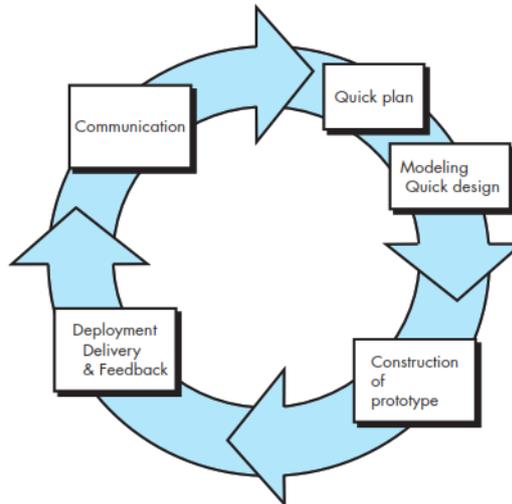
$dist[i][k]$ adalah jarak terpendek antara node i dan k

$dist[k][j]$ adalah jarak terpendek antara node k dan j

Algoritma ini memiliki kompleksitas waktu $O(n^3)$, di mana n adalah jumlah node pada grafik.

2.3 Metode Pengembangan Perangkat Lunak

Prototype merupakan metode yang efektif dalam merancang perangkat lunak. Prototype dimulai dengan mengumpulkan kebutuhan yang akan di rancang. Pengembangan mendefinisikan objek keseluruhan dari perangkat lunak, mengidentifikasi segala aktifitas yang diketahui dan kemudian melakukan “perancangan kilat”. Perancangan kilat berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan atau pemakai (contohnya pendekatan input dan format output) (Roger S. Pressman and Bruce R. Maxim 2015). Gambar 2.1 berikut adalah metode prototype :



Gambar 2.1 Metode Prototype

Tahapan-tahapan dalam metode Prototype adalah sebagai berikut :

1. Komunikasi (Communication) dan pengumpulan data awal, yaitu komunikasi dengan klien dan user untuk menentukan kebutuhan.
2. Perencanaan cepat (Quick Plan), yaitu pembuatan perencanaan analisis terhadap kebutuhan pengguna.
3. Pemodelan perancangan cepat (Modeling Quick Design), yaitu membuat rancangan desain program.
4. Pembentukan prototype (Construction of prototype), yaitu pembuatan aplikasi berdasarkan dari pemodelan desain yang telah dibuat.
5. Penyerahan sistem dan umpan balik (Development Delivery and Feedback), yaitu memproduksi perangkat secara benar sehingga dapat digunakan oleh pengguna.

2.7 *Unified Modeling Language (UML)*

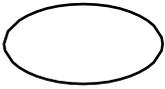
Menurut (R. A. Sukanto and Shalahuddin 2013) menjelaskan bahwa “UML (Unified Modeling Language) adalah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun menggunakan teknik pemograman berorientasi objek. UML (Unified

Modeling Language) muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasi, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak.”

1. Use Case Diagram

(John W Satzinger, Robert B Jackson 2005) menguraikan bahwa “Use Case Diagram merupakan rangkaian tindakan yang dilakukan oleh sistem, aktor mewakili user atau sistem lain yang berinteraksi dengan sistem yang dimodelkan.” Tabel 2.2 berikut merupakan simbol-simbol use case diagram disertai dengan penjelasan kegunaan dari simbolnya :

Tabel 2. 1 Simbol Use Case Diagram

Gambar	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Generalization</i>	Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor).
<<Include>>	<i>Include</i>	Menspesifikasikan bahwa use case sumber secara <i>eksplisit</i> .
<<Extend>>	<i>Extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.

	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2. 1 diatas merupakan Use Case Diagram menjelaskan simbol-simbol yang digunakan untuk membuat use case diagram beserta nama dari setiap simbol dan dilengkapi dengan penjelasan kegunaan dari setiap simbol. Dengan demikian dapat disimpulkan bahwa use case diagram digunakan untuk mengetahui alur atau tindakan yang akan terjadi pada sebuah aplikasi yang akan kita bangun. Use case diagram menggambarkan bagaimana user atau aktor akan berinteraksi dengan aplikasi yang dibangun.

2. Activity Diagram

(R. A. Sukanto and Shalahuddin 2013) menguraikan bahwa “diagram aktivitas atau activity diagram adalah menggambarkan aliran kerja atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktifitas menggambarkan aktifitas sistem bukan apa yang dilakukan oleh aktor.”

Tabel 2. 2 Simbol Activity Diagram

Gambar	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmukasaling berinteraksi satu sama lain.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan

	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran
---	------------------	--

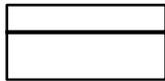
Tabel 2.2 tersebut diatas merupakan Simbol Activity Diagram menjelaskan simbol-simbol yang digunakan untuk membuat activity diagram beserta nama dari setiap simbol dan dilengkapi dengan penjelasan kegunaan dari setiap simbol. Dengan demikian dapat disimpulkan bahwa activity diagram adalah penjelasan tentang alur kerja dari setiap menu yang ada pada program. Menggambarkan bagaimana sebuah menu akan bereaksi. Pada Diagram Activity lebih menekankan pada penjelasan alur atau aktifitas menu dari aplikasi bukan aktifitas yang dilakukan oleh user atau aktor.

3. Class Diagram

(R. A. Sukanto and Shalahuddin 2013) menguraikan bahwa “diagram kelas atau class diagram menggambarkan struktur program dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun aplikasi.”

Tabel 2. 3 Simbol Class Diagram

Gambar	Nama	Keterangan
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.

	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

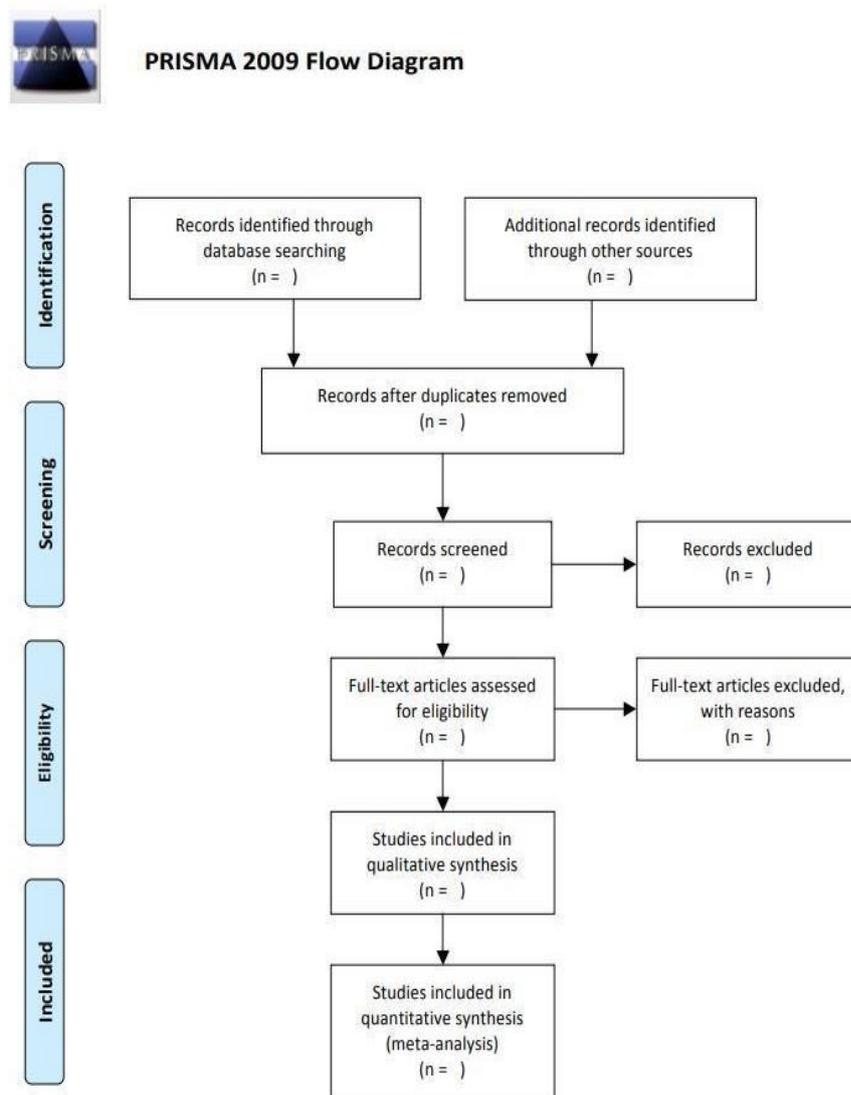
Tabel 2.3 merupakan Simbol Class Diagram menjelaskan simbol-simbol yang digunakan untuk membuat class diagram beserta nama dari setiap simbol dan dilengkapi dengan penjelasan kegunaan dari setiap simbol. Dengan demikian dapat disimpulkan bahwa class diagram adalah menjelaskan tentang penggambaran struktur dari program aplikasi yang akan dibangun.

2.4 *Black Box Testing*

Black box testing merupakan teknik pengujian yang melakukan pendekatan kebutuhan dasar program dalam mengecek fungsional program (Swastika & Putra, 2016:73). Menurut Sukamto & Shalahuddin (2015:275) mendefinisikan bahwa “black box testing yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan”. Berdasarkan penjelasan diatas mengenai Pengujian unit (black box testing) dapat disimpulkan bahwa black box testing adalah suatu pengujian program yang 24 dibuat bertujuan untuk menemukan kesalahan yang tidak diduga serta mengatasi penyelesaian penemuan kesalahan yang terjadi.

2.5 PRISMA Diagram

PRISMA (*Preferred Reporting Items for Systematic Reviews And Meta Analysis*) merupakan alat dan panduan yang digunakan untuk melakukan penelitian terhadap sebuah *systematic reviews* dan *meta analysis*. Prisma diagram adalah diagram alir menggambarkan aliran informasi melalui fase yang berbeda dari tinjauan sistematis dengan memetakan jumlah catatan yang diidentifikasi, disertakan dan dikecualikan, dan alasan pengecualian (Page et al. 2021)



Gambar 2.1 Diagram PRISMA

Gambar 2.1 di atas dapat disimpulkan bahwa pembuatan *Systematic Literature Review* (PRISMA) terdiri dari 4 langkah, yaitu:

1. *Identification*

Identification (identifikasi) adalah kegiatan yang bertujuan untuk memeriksa dan menganalisa, dalam proses ini peneliti mengidentifikasi masalah penelitian (*research problem*) dikarenakan hal tersebut menentukan kualitas suatu penelitian. Dalam penelitian ini peneliti mengkaji permasalahan melalui banyak jurnal penelitian yang bersumber dari google cendekia. Kata kunci dari penelitian ini adalah penerapan algoritma *floyd warshall*.

2. *Screening*

Screening (penyaringan) adalah penyaringan pemilihan data yang bertujuan untuk memilih masalah penelitian yang sesuai dengan topik yang diteliti. Adapun judul yang diteliti dalam penelitian ini adalah penerapan algoritma *floyd warshall*. Dengan topik tersebut, data jurnal yang diakses dalam proses penelitian ini di-*screening* berdasarkan pada kriteria sebagai berikut :

- a. Jurnal diterbitkan dalam rentang waktu tahun 2018-2022
- b. Tipe jurnal *research articles*
- c. Jurnal yang dapat diakses secara penuh
- d. Pengecekan berdasarkan duplikasi (kesamaan jurnal)

3. *Eligibility*

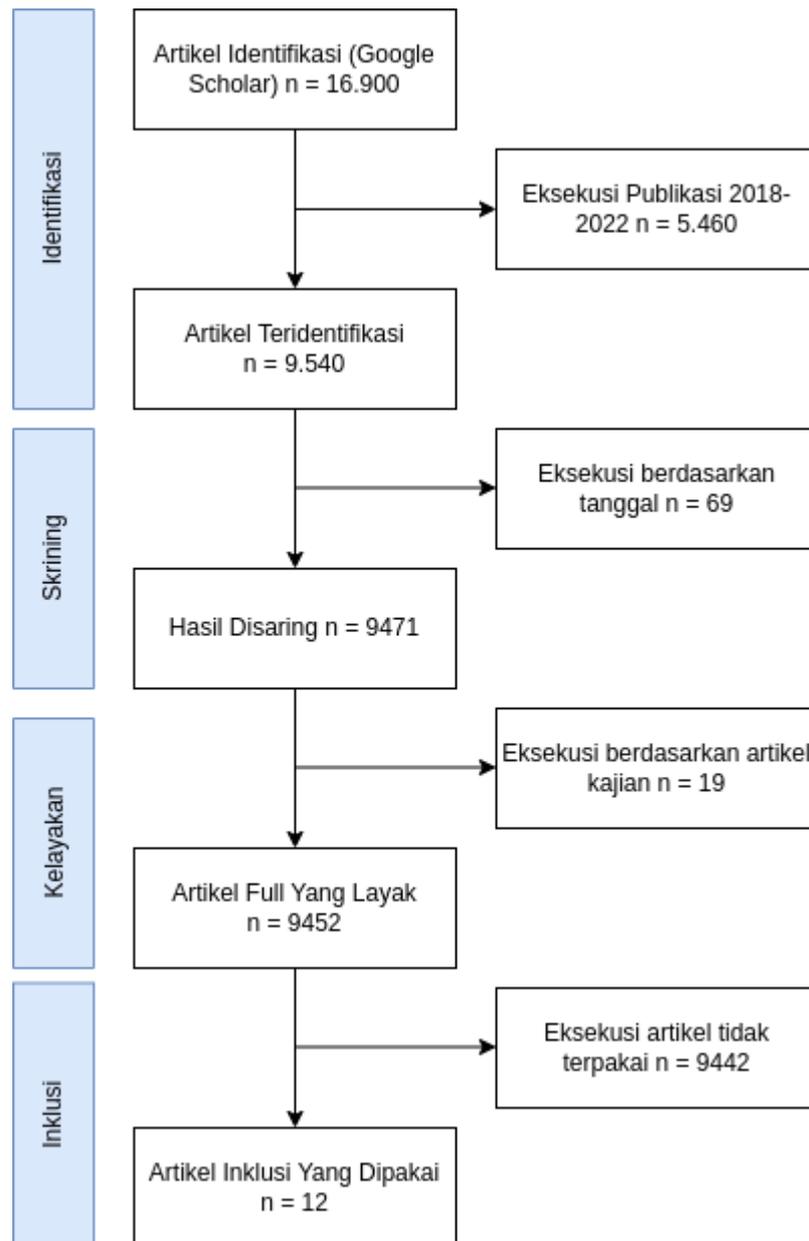
Tahap *eligibility* adalah tahap penentuan kelayakan artikel, peneliti menentukan artikel yang akan dijadikan bahan *literature review* yang telah menentukan kelayakan yaitu artikel harus berdasarkan riset asli dan artikel memiliki bahasan mengenai "*Floyd Warshall*".

4.Included

Included merupakan tahap terakhir dari PRISMA (*Preferred Reporting Items for Systematic Reviews And Meta-Analyses*) *flow diagram*, pada tahap ini bertujuan untuk pengolahan data. Setelah artikel tersebut memenuhi kelayakan maka akan didapatkan jumlah artikel yang akan diteliti.

2.6 Tahapan Systematic Review

Tahapan-tahapan *systematic review* yang harus dilakukan sehingga hasil dari studi literatur tersebut dapat diakui kredibilitasnya. Adapun tahapan-tahapan tersebut digambarkan pada Gambar 2.2



Gambar 2.2 Diagram PRISMA

Literatur-literatur review pada penelitian ini, peneliti memasukkan kata kunci “ *floyd warshall* ” dari Google Scholar dan melakukan beberapa tahapan yaitu identifikasi masalah yang dalam tahap ini peneliti mengumpulkan artikel sebanyak 16.900 data yang berfokus pada algoritma *floyd warshall* seperti Gambar 2.3

Google Cendekia floyd warshall

Artikel Sekitar 16.900 hasil (0,05 detik)

Kapan saja
Sejak 2023
Sejak 2022
Sejak 2019
Rentang khusus...

Urutkan menurut relevansi
Urutkan menurut tanggal

Semua jenis
Artikel kajian

sertakan paten
 mencakup kutipan
 Buat notifikasi

The Floyd-Warshall algorithm on graphs with negative cycles
S. Hugarey - Information Processing Letters, 2010 - Elsevier
... in $O(m)$ and to start the **Floyd-Warshall** algorithm only if the input graph has no negative cycles. Most implementations of the **Floyd-Warshall** algorithm we have seen apply neither of ...
☆ Simpan 90 Kutip Dirujuk 234 kali Artikel terkait 7 versi

Floyd-warshall algorithm
EW Weisstein - https://mathworld.wolfram.com/, 2008 - mathworld.wolfram.com
The **Floyd-Warshall** algorithm, also variously known as **Floyd's** algorithm, the **Roy-Floyd** algorithm, the **Roy-Warshall** algorithm, or the **WFI** algorithm, is an algorithm for efficiently and ...
☆ Simpan 90 Kutip Dirujuk 27 kali Artikel terkait 80

On the Floyd-Warshall algorithm for logic programs
C. Papadimitriou, M. Sideri - The journal of logic programming, 1999 - Elsevier
We explore the possibility of evaluating single-rule Datalog programs efficiently and with logarithmic work space by a natural extension of the **Floyd-Warshall** algorithm for transitive ...
☆ Simpan 90 Kutip Dirujuk 44 kali Artikel terkait 9 versi

Speeding up the Floyd-Warshall algorithm for the cycled shortest path problem
A. Anis, A. Saleh - Applied Mathematics Letters, 2012 - Elsevier
... We review the **Floyd-Warshall** algorithm that finds both the ... effort of the **Floyd-Warshall** algorithm substantially. Besides, the ... algorithms, especially the **Floyd-Warshall** algorithm, which ...
☆ Simpan 90 Kutip Dirujuk 95 kali Artikel terkait 5 versi

Gambar 2.3 Jumlah Hasil Pencarian *Floyd Warshall*

Tahap ini juga dilakukan eksekusi publikasi berdasarkan tahun 2018-2022 dengan hasil teridentifikasi 5.460 data artikel seperti Gambar 2.4

Google Cendekia floyd warshall

Artikel Sekitar 5.460 hasil (0,04 detik)

Kapan saja
Sejak 2023
Sejak 2022
Sejak 2019
Rentang khusus...

2018 — 2022
Telusuri

Urutkan menurut relevansi
Urutkan menurut tanggal

Semua jenis
Artikel kajian

sertakan paten
 mencakup kutipan
 Buat notifikasi

Prim and floyd-warshall comparative algorithms in shortest path problem
Z. Ramadhani, A. U. Sabhan - Proceedings of the Joint ... 2018 - books.google.com
... is titled "Performance analysis of **Floyd-Warshall** algorithms vs. rectangular algorithm." In this paper, they have examined the comparative study of the **Floyd Warshall** algorithm and the ...
☆ Simpan 90 Kutip Dirujuk 28 kali Artikel terkait 5 versi

Perbandingan Algoritma Prim dengan Algoritma Floyd-Warshall dalam Menentukan Rute Terpendek (Shortest Path Problem)
Z. Ramadhani, M. Zaitis, S. Efendi - JURIKOM (Jurnal ... 2018 - ejournal.stmik-budidarma.ac.id
Masalah optimasi menjadi hal yang kompleks dalam mencari jalur atau rute optimal, banyak metode yang menjadi indikator rute optimal salah satunya adalah rute terpendek. ...
☆ Simpan 90 Kutip Dirujuk 43 kali Artikel terkait 5 versi 80

Application of optimization heavy traffic path with floyd-warshall algorithm
IKLD Pandika, B. E. W. - Conference on Control ... 2018 - ieeexplore.ieee.org
... In this research, **Floyd-Warshall** algorithm is used to be able to optimize the use of toll road ... This application uses **FloydWarshall** Algorithm because the advantages of this algorithm ...
☆ Simpan 90 Kutip Dirujuk 12 kali Artikel terkait

Comparison of Floyd-Warshall algorithm and Greedy Algorithm in determining the shortest route
H. Agus, D. Lanting, Y. Saito - 2018 2nd East Indonesia ... 2018 - ieeexplore.ieee.org
... **FloydWarshall** algorithm whereas for the Heuristic method, the greedy algorithm is employed. The **Floyd-Warshall** ... final result obtained is the **FloydWarshall** algorithm provides a better ...
☆ Simpan 90 Kutip Dirujuk 16 kali Artikel terkait

Gambar 2.4 Hasil Penyaringan Data Tahun 2018-2022

Screening yang dalam tahap ini peneliti mengumpulkan artikel sebanyak 69 data yang telah dilakukan tahap penyaringan dan memiliki fokus kajian algoritma *floyd warshall* berdasarkan tanggal kajian seperti Gambar 2.5

Google Cendekia "floyd warshall"

Artikel Sekitar 69 hasil (0,04 dik)

Kapan saja
Sejak 2023
Sejak 2022
Sejak 2019
Rentang khusus...

Urutkan menurut relevansi
Urutkan menurut tanggal

Semua jenis
Artikel kajian

sertakan paten
 mencakup kutipan

Buat notifikasi

Artikel-artikel yang ditambahkan pada tahun lalu, diurutkan berdasarkan tanggal

Rancang Bangun Aplikasi Pemandu Wisata Backpacker Menggunakan Algoritma Floyd-Warshall [PDF] upnjatim.ac.id
MN Pratama - 2023 - repository.upnjatim.ac.id
7 hari lalu - ... Bangun Aplikasi Pemandu Wisata Backpacker Menggunakan Algoritma Floyd-Warshall. Penerapan Algoritma floyd-warshall pada aplikasi berbasis Android ini dapat ...
☆ Simpan 99 Kutip 2 versi 88

SISTEM INFORMASI GEOGRAFIS BERBASIS WEB MENGGUNAKAN ALGORITMA FLOYD WARSHALL UNTUK PEMETAAN RUMAH TAHFIDZ AL-QURAN DI KOTA ...
FM Taufiqy, I Irfaza, M Alida - Syntax: Journal of ..., 2022 - jurnal.dharmawangsa.ac.id
23 hari lalu - ... Peneliti menerapkan algoritma Floyd Warshall sebagai fitur pencarian rute terdekat pada ... Algoritma Floyd Warshall berhasil diimplementasikan ke dalam sistem, dan dapat ...
☆ Simpan 99 Kutip 88

ИСПОЛЬЗОВАНИЕ АЛГОРИТМОВ НА ГРАФАХ ДЛЯ ПРИКЛАДНЫХ ЗАДАЧ МИНИМИЗАЦИИ ПРОИЗВОДСТВЕННЫХ ЗАТРАТ
ДО Григорьева - АКТУАЛЬНЫЕ ВОПРОСЫ СОВРЕМЕННОЙ НАУКИ ..., 2022 - elibrary.ru
23 hari lalu - В статье рассматриваются методы представления графов в памяти электронных вычислительных машин, циклы в графах, алгоритмы поиска кратчайших путей в ...
☆ Simpan 99 Kutip

Gambar 2.5 Hasil Penyaringan Berdasarkan Tanggal Kajian

Tahapan setelah screening menghasilkan artikel kajian yang layak sebanyak 19 data seperti Gambar 2.6

Google Cendekia "floyd warshall"

Artikel Sekitar 19 hasil (0,05 dik)

Kapan saja
Sejak 2023
Sejak 2022
Sejak 2019
Rentang khusus...

Urutkan menurut relevansi
Urutkan menurut tanggal

Semua jenis
Artikel kajian

Buat notifikasi

Artikel-artikel yang ditambahkan pada tahun lalu, diurutkan berdasarkan tanggal

A Comprehensive Survey of Graph-level Learning [PDF] arxiv.org
Z Yang, G Zhang, J Wu, J Yang, QZ Sheng - arXiv preprint arXiv ..., 2023 - arxiv.org
9 hari lalu - ... For example, the shortest-path kernel [23] determines the shortest path between the vertices v and u via the Floyd-Warshall [53] or Dijkstra's [54] algorithms. The distance ...
☆ Simpan 99 Kutip 88

Hierarchical deposition and scale-free networks: A visibility algorithm approach
J Berez - Physical Review E, 2022 - APS
45 hari lalu - ... It can be shown numerically (see Appendix C), by means of a Floyd-Warshall algorithm [61], that the network diameter is again a power law of the number of vertices, ie, $D(N) \dots$
☆ Simpan 99 Kutip 3 versi

A Review of Shortest Path Problem in Graph Theory [PDF] geniusjournals.org
K Nosirov, E Norov, S Tashmetov - Eurasian Journal of ..., 2022 - geniusjournals.org
49 hari lalu - ... Floyd-Warshall algorithm [52]. Although their system can recognize negative-weight cycles, it cannot break them. FloydWarshall ... Because the Floyd-Warshall technique does ...
☆ Simpan 99 Kutip 2 versi 88

Reinforcement Learning-Based Traffic Engineering in SDN: Problem Formulation, Parameters, and Topology
A Bhavani, Y Ekshitha, A Mounika, U Prabhu - ... : Proceedings of ICICCT ..., 2022 - Springer
69 hari lalu - ... The Floyd-Warshall algorithm is employed for all couples to solve the shortest path ... The Floyd-Warshall method is used to find all pair shortest path problems from a given ...
☆ Simpan 99 Kutip 2 versi

Gambar 2.6 Hasil Penyaringan Kelayakan Artikel

Data-data dari 19 (sembilan belas) data tersebut merupakan artikel yang tidak dipakai sebanyak 3 data dan artikel inklusi yang dipakai sebanyak 12 data sebagaimana terlihat pada Tabel 2.4

Tabel 2.4 Penelitian Terdahulu

No	Judul, Penulis, Tahun	Deskripsi	Algoritma	Search Engine	Open Source dataset	Akurasi
1	InfoTekJar Algoritma Floyd Warshall Dan Colaborative Filtering Untuk Penentuan Rekomendasi Dan Rute Terpendek Pencarian Apotik: Studi Eksperimen (Muzakir 2020) http://eprints.binadarma.ac.id/13377/	Aplikasi ini membahas tentang rute terpendek dan rekomendasi pencarian apotik. Pada jurnal ini, data yang digunakan adalah apotik yang berada di kota Palembang,	<i>Floyd Warshall, Colaborative Filtering</i>	Google Scholar	Kimia Farma Dempo, Adhithia, K24 Wahid Hasyim, Guardian	-
2	Jurnal Nasional Teknik Elektro dan Teknologi Informasi Penerapan Floyd-Warshall Untuk Pencarian Rute Terpendek Pada Aplikasi Notifikasi Kecelakaan Lalu Lintas (Haniah Mahmudah et al. 2022) https://journal.ugm.ac.id/v3/JNTETI/article/view/2201	Aplikasi ini digunakan untuk mendeteksi kecelakaan lalu lintas realtime dengan menggunakan algoritma Floyd Warshall, sehingga dengan menggunakan algoritma tersebut dapat mencari lokasi terdekat untuk sampai ke tempat kecelakaan. Dan, di aplikasi ini terdapat kantor polisi dan rumah sakit untuk membantu korban kecelakaan.	<i>Floyd Warshall</i>	Google Scholar	-	100%
3	JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika) Implementasi <i>Bellman-Ford</i> dan <i>Floyd Warshall</i> Dalam Menentukan Jalur Terpendek Menuju Universitas Nasional Berbasis Android (Farhan, Andryana, and Hayati 2020) http://www.jurnal.stkipggritulungagung .	Aplikasi ini menggunakan 2 algoritma yaitu Bellman-Ford dan Floyd warshall, kedua nya memiliki kesamaan yaitu dapat menentukan jalur terdekat untuk menuju universitas nasional.	<i>Floyd Warshall. Bellman-Ford</i>	Google Scholar	-	39.40% dan 25,24%

	ac.id/index.php/jipi/article/view/1812/841					
4	<p>Techno.com (Jurnal Teknologi Informasi) Penentuan Rute Terpendek Pendistribusian Barang Menggunakan Algoritma Floyd Warshall (Herlambang, Fauzan, and Fathonah 2021)</p> <p>http://publikasi.dinus.ac.id/index.php/technoc/article/view/4686</p>	<p>Aplikasi ini membantu mendistribusikan barang dengan jalur terpendek dari PT. Pos Logistik Jakarta dan PT. Pos Logistik Tambun. Penelitian ini diharapkan agar bermanfaat bagi perusahaan agar tingkat keterlambatan pada proses pengiriman barang dapat berkurang.</p>	<i>Floyd Warshall</i>	Google Scholar	PT. Pos Logistik Jakarta dan PT. Pos Logistik Tambun	-
5	<p>JTSI (Jurnal Teknologi Sistem Informasi) Studi Perbandingan Metode Floyd-Warshall dan Haversine Untuk Pencarian Lokasi Panti Asuhan Dikota Palembang Lewat Aplikasi Berbasis Android (Pribadi, Puspasari, and Gustriansyah 2022)</p> <p>https://jurnal.mdp.ac.id/index.php/jtsi/article/view/2452/748</p>	<p>Aplikasi ini berisi tentang informasi Panti Asuhan yang ada di Kota Palembang. Aplikasi ini digunakan sebagai sarana informasi lengkap panti asuhan dikarenakan banyaknya masyarakat yang masih kesulitan untuk menemukan informasi yang lengkap untuk Panti Asuhan yang ada di Kota Palembang</p>	<i>Floyd Warshall,, Haversine</i>	Google Scholar	Panti Asuhan Palembang	UAT 84,6%
6	Journal Basic Science And Technology Rancang Bangun Aplikasi Web	<p>Aplikasi ini dibuat untuk mendapatkan rute terpendek dari gedung ke gedung</p>	<i>Floyd Warshall</i>	Google Scholar	Universitas Brawijaya	100%

	<p>Pencarian Rute Terpendek Antar Gedung Di Kampus Menggunakan Algoritma Floyd Warshall (Fanani, Adams J, and Wicaksono 2012)</p> <p>https://www.researchgate.net/profile/Lutfi-Fanani/publication/292971032_Rancang_Bangun_Aplikasi_Web_Pencarian_Rute_Terpendek_Antar_Gedung_di_Kampus_Menggunakan_Algoritma_Floyd-warshall/links/5db55abc299bf111d4d04e8d/Rancang-Bangun-Aplikasi-Web-Pencarian-Rute-Terpendek-Antar-Gedung-di-Kampus-Menggunakan-Algoritma-Floyd-warshall.pdf</p>	<p>yang ada di kampus Universitas Brawijaya. Hasil pengujian aplikasi ini menunjukkan bahwa keseluruhan fungsional bekerja dengan baik, dengan tingkat akurasi 100%</p>				
7	<p>Jurnal Pengembangan Teknologi Informasi Dan Komputer Rekomendasi Tempat Wisata Kota Malang Menggunakan Metode <i>Profile Matching</i> Dan Saran Route Menggunakan <i>Floyd Warshall</i> Berbasis Android (Chairi et al. 2018)</p> <p>https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/1523/517</p>	<p>Aplikasi ini dibuat untuk para wisatawan yang berkunjung ke Kota Malang untuk menemukan Tempat Wisata yang ada. Dengan menggunakan Algoritma Floyd Warshall, para wisatawan dapat mencari rute terpendek ke tempat wisata.</p>	<i>Floyd Warshall, Profile Matching</i>	Google Scholar	Aset Wisata Kota Malang	60% CF, 40% SF, Akurasi 75%
8	<p>Jurnal Ilmiah Core IT Implementasi Algoritma Floyd Warshall Dalam Perancangan Aplikasi Jareno (Jasa Renovasi) (Edi and Tarigan 2022)</p>	<p>Aplikasi ini mampu memberikan rute yang paling optimal untuk para penyedia Jasa Renovasi untuk menuju ke lokasi pelanggan agar dapat memberikan pelayanan yang sangat cepat.</p>	<i>Floyd Warshall</i>	Google Scholar	-	-

	http://ijcoreit.org/index.php/coreit/article/view/366					
9	<p>JUST IT (Jurnal Sistem Informasi, Teknologi Informasi dan Komputer Pencarian Lokasi Apotek Terdekat Menggunakan Algoritma Floyd Warshall (Mujiastuti and Purwanto 2018)</p> <p>https://jurnal.umj.ac.id/index.php/just-it/article/view/3231/2503</p>	<p>Aplikasi ini memberikan informasi apotek yang ada di daerah Jakarta Timur. Di Dalam aplikasi ini, pengguna dapat melihat informasi peta, lokasi user dan apotek, pencarian apotek, daftar apotek terdekat pada lokasi pengguna, disertai no telp dan rute jalan menuju apotek.</p>	<i>Floyd Warshall, Haversine Formula</i>	Google Scholar	-	-
10	<p>SNIA (Seminar Nasional Informatika dan Aplikasinya) Sistem Pendukung Keputusan Penentuan Rute Terbaik Menuju Posko Pendakian Gunung Ciremai Menggunakan Algoritma Floyd Warshall Dan Weighted Product (Heni Sulastrri, Husni Mubarak 2019)</p> <p>https://snia.unjani.ac.id/web/index.php/snia/article/view/192</p>	<p>Aplikasi ini dapat membantu pengguna untuk mencari rute terpendek untuk menuju ke Posko Pendakian Gunung Ciremai. Output dari penelitian ini merupakan sistem yang dapat memberikan rekomendasi atau alternatif pilihan rute terbaik menuju posko pendakian Gunung Ciremai dengan nilai bobot kriteria (w: 33, 30, 16, 16, 50), akurasi ketepatan manual dan program adalah 100%.</p>	<i>Floyd Warshall, Weighted Product</i>	Google Scholar	Posko Pendakian Gunung Ciremai	Ketepatan Manual dan Program : 100%
11	<p>JEBE (Journal Of Economic, Business and Engineering Penerapan Algoritma Floyd Warshall Dalam Menentukan Rute Terpendek Pada Pencarian Pengepul Sayur Di Kecamatan Kepil (Heni Sulastrri, Husni Mubarak 2019)</p> <p>https://ojs.unsiq.ac.id/index.php/jebe/article/view/2725</p>	<p>Aplikasi ini dibuat untuk mempermudah petani dalam mencari pengepul sayur yang ada di Kecamatan Kepil. Dengan menggunakan Algoritma Floyd Warshall, para petani untuk menentukan jarak terdekat untuk sampai ke pengepul sayur yang ada di Kecamatan Kepil</p>	<i>Floyd Warshall</i>	Google Scholar	Kecamatan Kepil	-

12	<p>Jurnal Sains, Aplikasi, Komputasi dan Teknologi Informasi</p> <p>Implementasi Algoritma Floyd Warshall Pada Pencarian Lokasi Agen Bus, Tour And Travel, Dan Rental Mobil Di Kota Palu Berbasis Android</p> <p>(Nugraha, Amriana, and Arif 2020)</p> <p>https://ocs.unmul.ac.id/index.php/jsakti/article/view/2400</p>	<p>Aplikasi ini menyediakan memberikan informasi lokasi agen bus, tour and travel, dan rental mobil yang ada di Kota Palu, dan sekaligus memberikan petunjuk jalur terpendek dengan menggunakan Algoritma Floyd warshall</p>	<p><i>Floyd Warshall</i></p>	<p>Google Scholar</p>	<p>-</p>	<p>-</p>
----	---	--	------------------------------	-----------------------	----------	----------