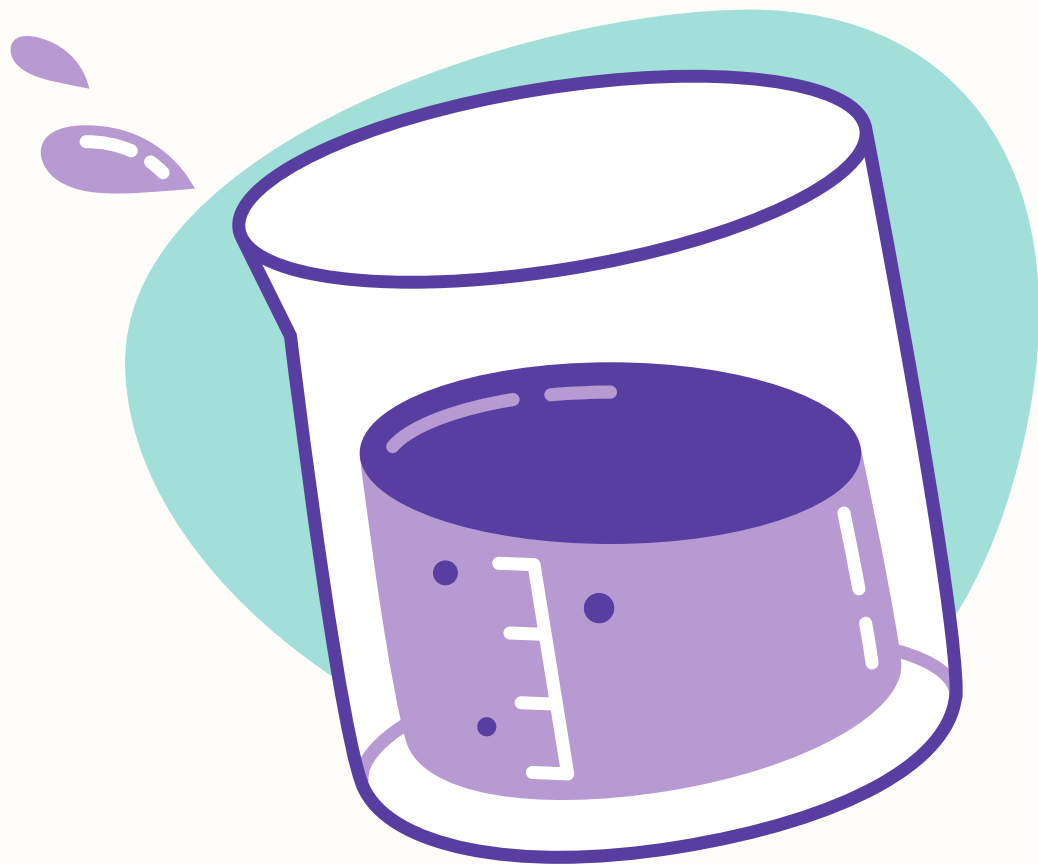


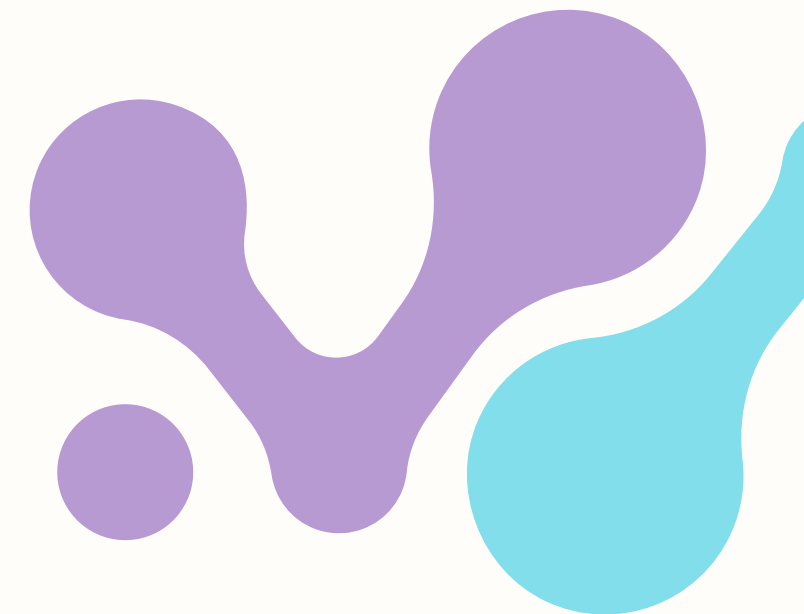
# RESEARCH



**PERBANDINGAN ALGORITMA A\* (STAR)  
DAN DIJKSTRA DALAM MENCARI RUTE  
TERCEPAT KE TEMPAT BERSEJARAH DI  
SURABAYA.**



**Muhammad Nova Ramadhana  
21081010235**

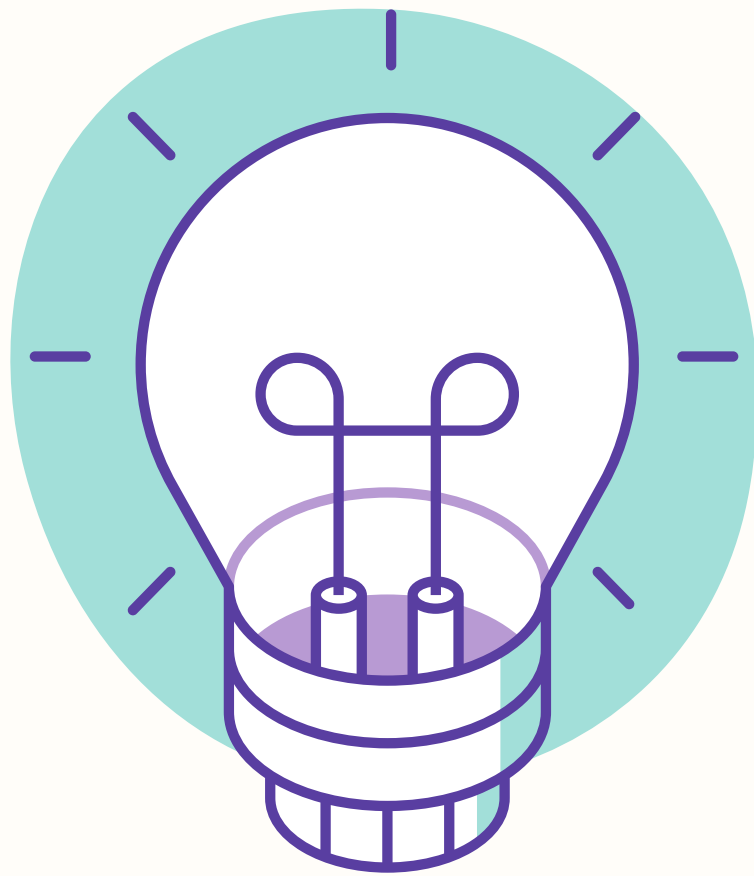


# Formulasi Permasalahan



## Latar Belakang:

- Tempat bersejarah di Surabaya menarik banyak wisatawan.
- Kendala utama: Kesulitan menentukan rute tercepat karena padatnya lalu lintas dan minimnya informasi.



## Rumusan Masalah:

1. Bagaimana performa algoritma A\* dibandingkan dengan algoritma Dijkstra dalam mencari rute tercepat di peta kota Surabaya menuju tempat bersejarah?
2. Faktor apa saja yang memengaruhi efisiensi kedua algoritma tersebut dalam konteks lingkungan geografis Surabaya?
3. Algoritma mana yang lebih optimal untuk digunakan dalam sistem navigasi berbasis lokasi di Surabaya?

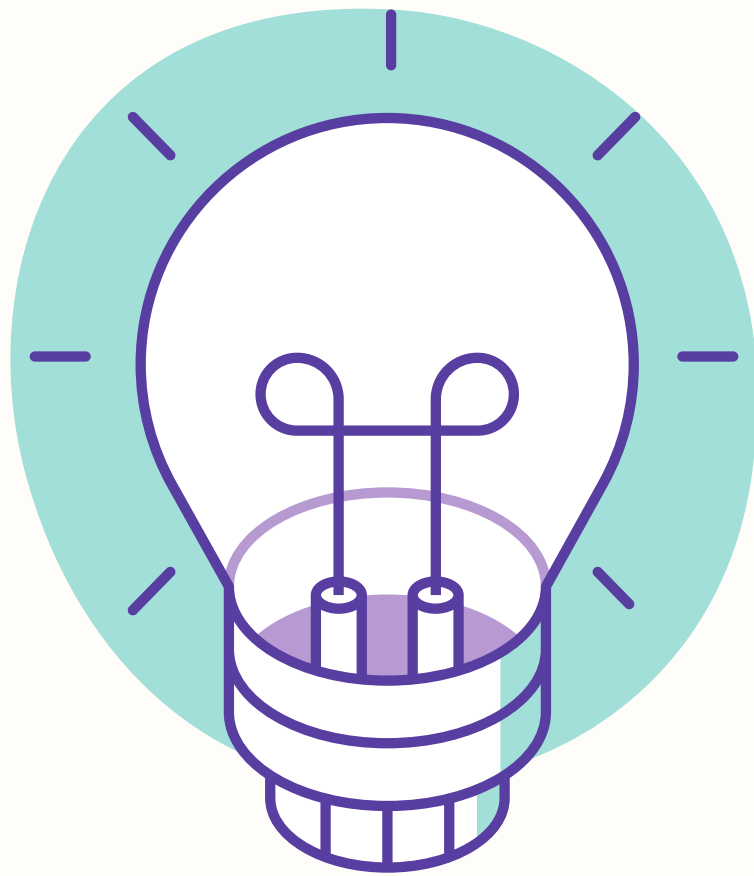


# Formulasi Permasalahan



## Research Gap:

1. Konteks Lokasi Spesifik: Sebagian besar penelitian sebelumnya hanya membandingkan algoritma A\* dan Dijkstra secara umum, tanpa mempertimbangkan karakteristik geografis spesifik seperti tata letak jalan, kepadatan lalu lintas, atau kondisi peta Surabaya.
2. Fokus pada Tempat Bersejarah: Tidak banyak penelitian yang secara khusus mengevaluasi algoritma pencarian rute tercepat menuju tempat bersejarah sebagai titik tujuan.
3. Kebutuhan Implementasi Praktis: Penelitian sebelumnya kurang memberikan panduan implementasi praktis pada aplikasi real-time yang dapat digunakan wisatawan atau masyarakat lokal.





# Mind Map

## Topik Utama: Perbandingan Algoritma Penentuan Rute Terpendek

Metode:

Pengumpulan data peta dan titik bersejarah

- Implementasi Algoritma Dijkstra & A\*
- Graf berbobot dari data Google Maps
- Pengujian dan evaluasi performa

Studi Terdahulu:

- Kota Toboali (Dijkstra untuk rute wisata berbasis web)
- Danau Toba (Dijkstra untuk rute wisata lokal)

Hasil dan Analisis

- Perbandingan waktu eksekusi
  - Akurasi hasil rute
- 

# Metode

## *Tahapan Penelitian*

- Pengumpulan Data:
  - a. Data peta Surabaya (OpenStreetMap, Google Maps, dll.)
  - b. Daftar tempat bersejarah di Surabaya.
- Pemodelan Peta:
  - Representasi graf berdasarkan simpul (nodes) dan tepi (edges).
- Implementasi Algoritma:
  - Implementasikan algoritma A\* dan Dijkstra menggunakan bahasa pemrograman (Python, Java, dll.).
- Simulasi dan Eksperimen:
  - Menalankan algoritma pada dataset peta dengan berbagai skenario.
- Evaluasi Hasil:
  - Bandingkan hasil berdasarkan waktu eksekusi, jumlah simpul yang diperiksa, dan efisiensi memori.

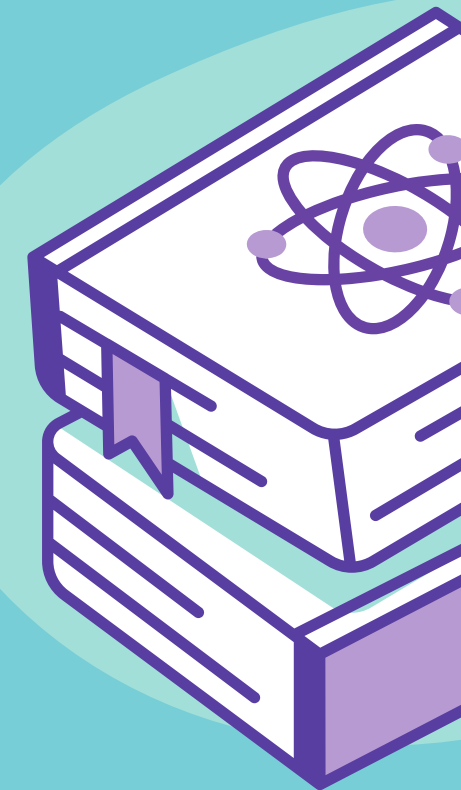
## *Parameter Uji*

- Waktu eksekusi algoritma.
- Jumlah simpul yang diperiksa.
- Panjang rute yang dihasilkan.
- Penggunaan memori.

# Matriks Pengujian

| Aspek Uji           | Parameter                       | Evaluasi   |
|---------------------|---------------------------------|--|
| Waktu Eksekusi      | Waktu (ms)                      | Stopwatch/perhitung<br>an internal program.        |
| Kompleksitas Proses | Jumlah simpul yang<br>diperiksa | Log data simpul selama<br>proses pencarian.        |
| Akurasi Rute        | Panjang Rute<br>(Km/m)          | Bandingkan Hasil<br>Rute dengan peta<br>sebenarnya |
| Efisiensi Memori    | Memori yang<br>digunakan (MB)   | Profiling memori<br>selama proses<br>pencarian     |





THANK YOU

