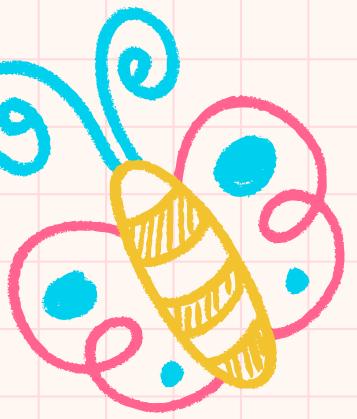


# BOOKSTORE NAVIGATION

Dimas Ananda D 5026241113

Avissa Riyandita P 5026241190

Izzana Firdaunnisa 5026241203

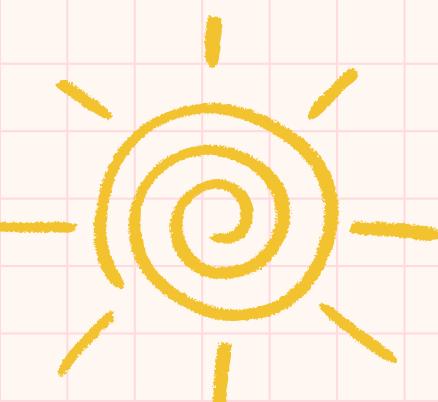


## BACKGROUND

Gramedia merupakan pusat perbelanjaan literatur dan informasi yang menawarkan puluhan ribu judul buku dari beragam kategori. Namun, banyak pengunjung mengalami kesulitan dalam menemukan rak atau lorong yang berisi kategori buku yang mereka butuhkan. Hal ini disebabkan oleh tata letak gramedia yang luas, kompleks, serta minimnya petunjuk lokasi yang membantu mahasiswa menavigasi lorong-lorong secara efisien.

### Problem :

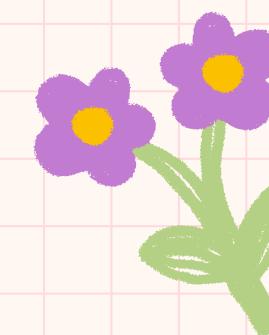
- Gramedia memiliki banyak lorong dengan kategori buku berbeda.
- Pengunjung sering kesulitan menemukan lokasi kategori buku yang mereka butuhkan.
- Dibutuhkan sistem yang dapat membantu menemukan lorong terdekat secara cepat.



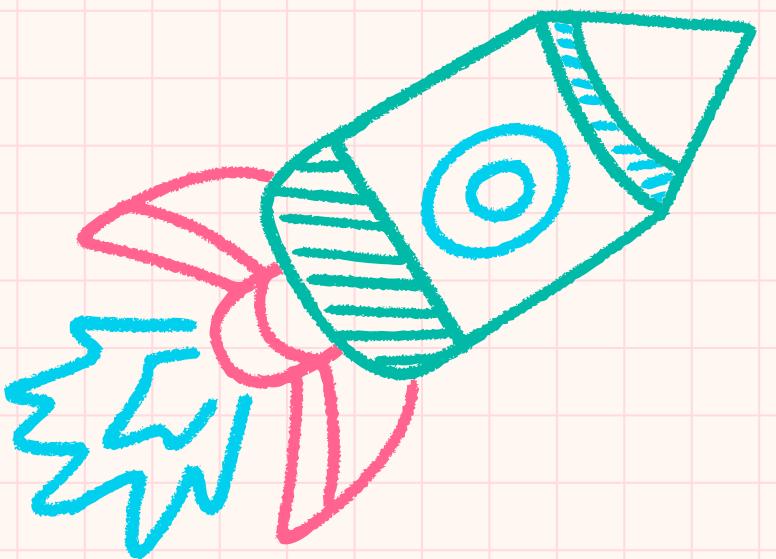


## TUJUAN

- Mempermudah Pengunjung Menemukan Buku berdasarkan kategori
- Mengoptimalkan Waktu Pencarian
- Membantu pengunjung menavigasi perpustakaan yang luas dan kompleks dengan lebih baik
- Mendukung Pencarian Kategori Buku Secara Sistematis

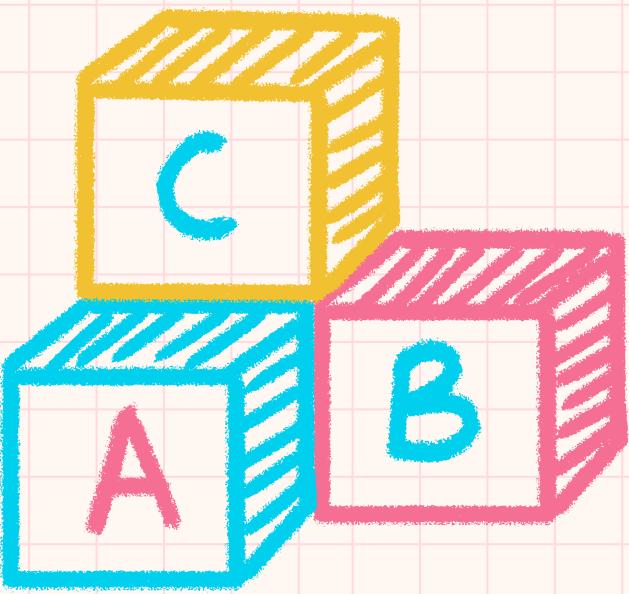


# ALGORITHM



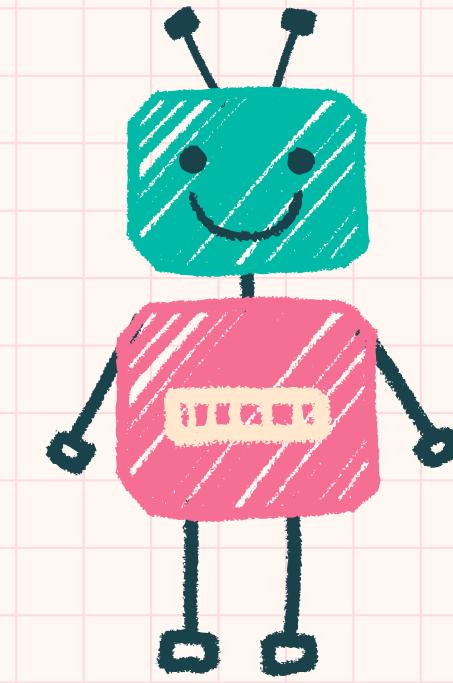
## GRAPH

Mempresentasikan kumpulan  
objek (node dan edge)



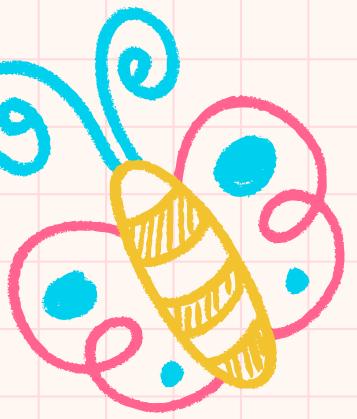
## BFS

Mencari jalur tercepat

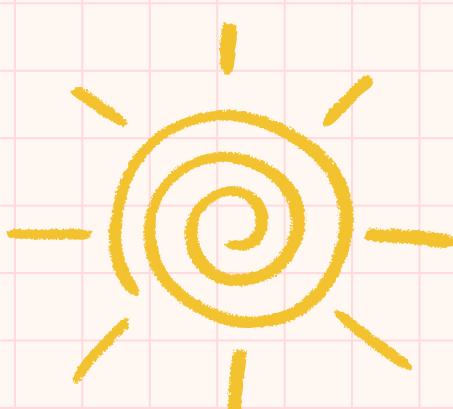
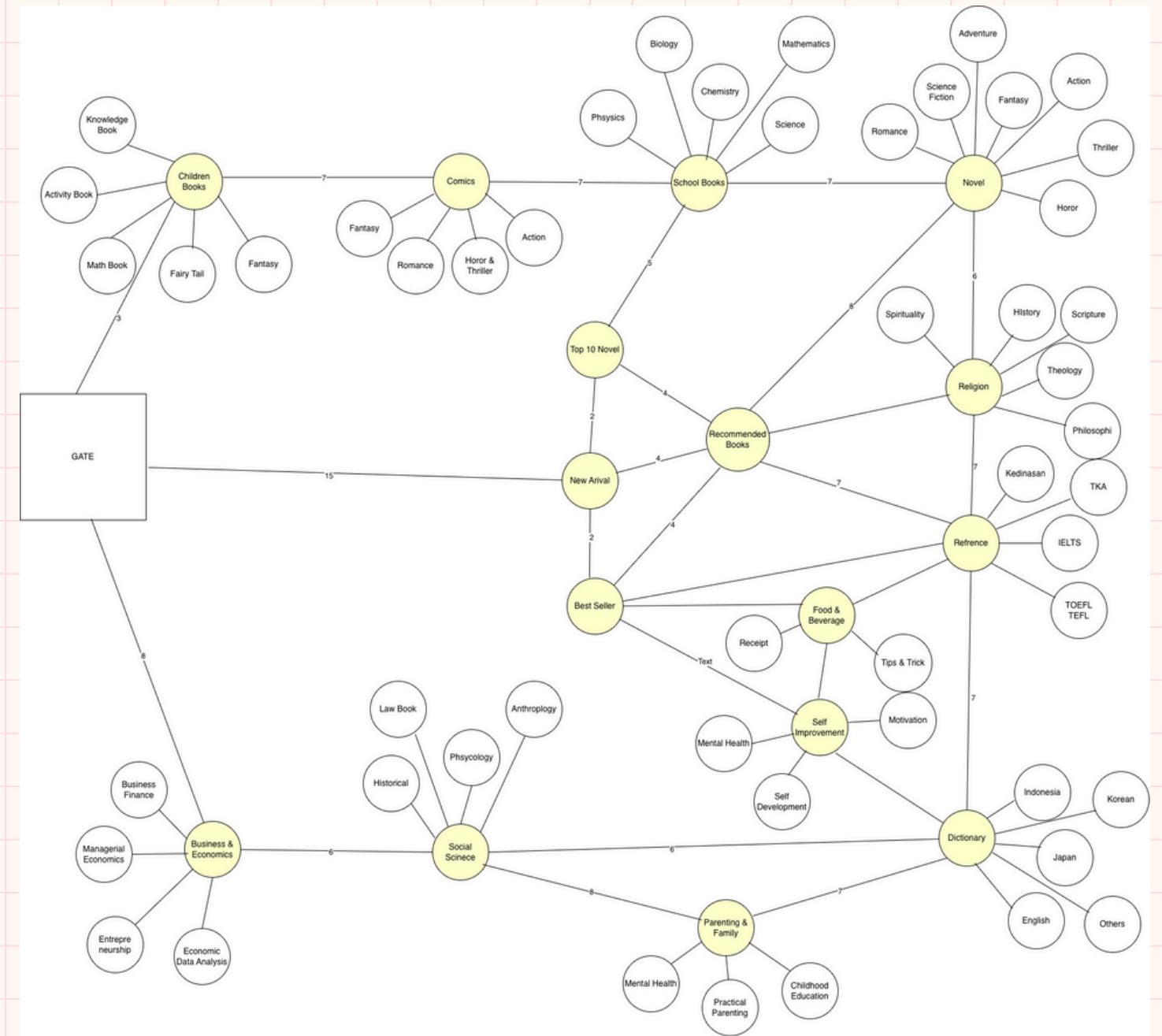


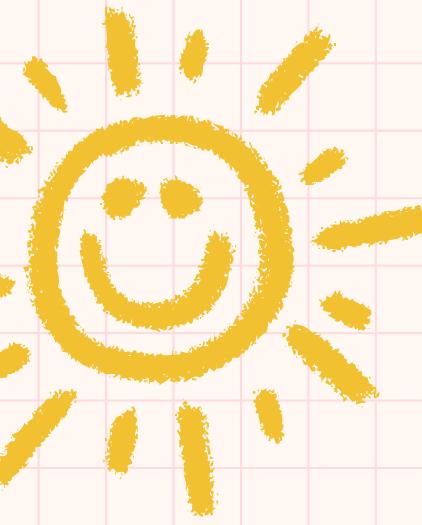
## DJIKSTRA

Mencari jalur terpendek



# DIAGRAM





# HOW SEMANTICS AFFECTS COMMUNICATION

Komponen utama:

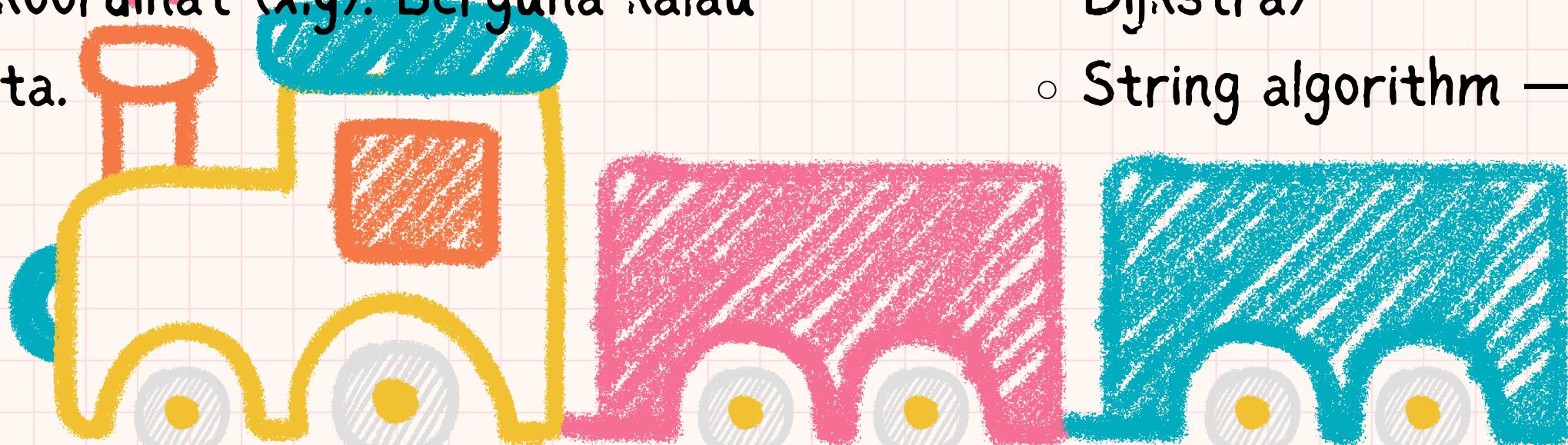
- BookNode (node graph)
- Point (koordinat visual)
- SearchResult (paket hasil)
- GramediaBookSearchSystem (graph + algoritma + UI CLI)
- Alur singkat: buat graph → pilih algoritma → cari → tampilkan hasil



# HOW SEMANTICS AFFECTS COMMUNICATION

Struktur data: BookNode & Point

- BookNode
  - String name (nama kategori)
  - List<String> connections (list tetangga)
  - Map<String,Integer> weights (bobot ke tiap tetangga)
  - Point coordinates (x,y untuk visualisasi)
  - method: addConnection(neighbor, weight)
- Point menyimpan koordinat (x,y). Berguna kalau ingin visualisasi peta.



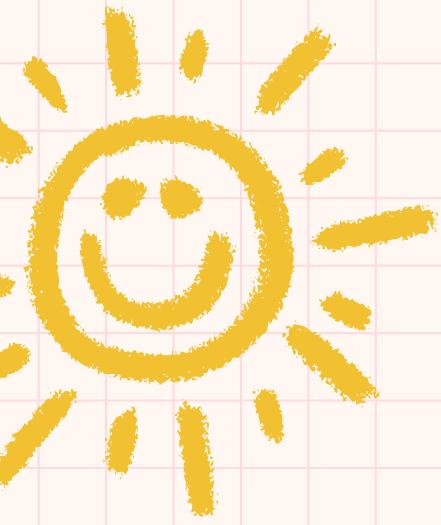
SearchResult (paket hasil)

- Fields:
  - List<String> path — jalur urut dari sumber ke target
  - int stepsCount — berapa banyak iterasi/kunjungan
  - boolean found — apakah target ditemukan
  - int totalDistance — total bobot (untuk Dijkstra)
  - String algorithm — "BFS" atau "Dijkstra"

# HOW SEMANTICS AFFECTS COMMUNICATION

buildGraph)

- addNode(name, x, y) — menambah node ke graph (HashMap)
- addEdge(from, to, weight) — menambah koneksi dua arah (menyimpan bobot)
- Contoh: addEdge("GATE", "Children Books", 5)
- Struktur mencakup:
  - Main categories (GATE, Novel, dsb.)
  - Sub-categories (Fantasy, Romance, dll.)
  - Koneksi antar node dengan bobot jarak/biaya



# HOW SEMANTICS AFFECTS COMMUNICATION

## BFS: Tujuan & Implementasi

- Fungsi: bfs(String start, String target)
- Pendekatan:
  - Gunakan Queue<List<String>> untuk menyimpan path sementara
  - visited untuk menghindari loop
  - Mulai dari start (GATE), perluas level-by-level
  - Jika menemukan target, kembalikan path
- Output SearchResult:
  - stepsCount = jumlah dequeue (node yang diproses)
  - totalDistance diset sebagai path.size()-1 (bukan bobot)



# HOW SEMANTICS AFFECTS COMMUNICATION

## BFS: Tujuan & Implementasi

- Fungsi: bfs(String start, String target)
- Pendekatan:
  - Gunakan Queue<List<String>> untuk menyimpan path sementara
  - visited untuk menghindari loop
  - Mulai dari start (GATE), perluas level-by-level
  - Jika menemukan target, kembalikan path
- Output SearchResult:
  - stepsCount = jumlah dequeue (node yang diproses)
  - totalDistance diset sebagai path.size()-1 (bukan bobot)



# THANK YOU

