

Algorithm A*:

The screenshot shows a Java project structure and a code editor for the `AStarAlgorithm.java` file.

Project Structure:

- Bot-AI_Side [route-optimizer] (D:\WSPA\eraser)
- .idea
- src
 - main
 - java
 - com.university.routing
 - algorithms
 - AStarAlgorithm
 - localSearch
 - TSPGeneticSolver
 - UseAlgorithms
 - Map
 - apiKey
 - DistanceMatrixService
 - DistanseMatrixResponse
 - GeocodingService
 - RoutelImage.java
 - models
 - Graph
 - Node
 - Main
 - org.example

AStarAlgorithm.java Content:

```
7      2 usages new *
8  public class AStarAlgorithm{
9      2 usages new *
10     public static List<String> findShortestPath(Graph graph, String start, String goal) {
11         PriorityQueue<Node> openSet = new PriorityQueue<>();
12         Set<String> closedSet = new HashSet<>();
13
14         Map<String, Integer> gCosts = new HashMap<>();
15         Map<String, String> cameFrom = new HashMap<>();
16
17         gCosts.put(start, 0);
18         openSet.add(new Node(start, gCost: 0, heuristic(start, goal)));
19
20         while (!openSet.isEmpty()) {
21             Node current = openSet.poll();
22
23             if (current.getId().equals(goal)) {
24                 return reconstructPath(cameFrom, goal);
25             }
26
27             closedSet.add(current.getId());
28
29             for (Map.Entry<String, Integer> neighbor : graph.getNeighbors(current.getId()).entrySet)
|               if (closedSet.contains(neighbor.getKey())) continue;
|               Node neighborNode = new Node(neighbor.getValue(), current.gCost + neighbor.getValue(),
|                                             heuristic(neighbor.getValue(), goal));
|               if (gCosts.getOrDefault(neighbor.getValue(), Integer.MAX_VALUE) > neighborNode.gCost) {
|                   gCosts.put(neighbor.getValue(), neighborNode.gCost);
|                   cameFrom.put(neighbor.getValue(), current.getId());
|                   openSet.add(neighborNode);
|               }
|           }
|       }
|   }
```