**Task 1:**

**Use any Framework/library, i.e.: Apache Cordova/Angular/React to implement HERE Map Display using HERE Javascript API and show a geocoded coordinate on the map based on an address input.**

**If framework/libraries are not desired, please implement the application with plain HERE Javascript API.**

**Link to HERE Javascript API: <https://developer.here.com/documentation/maps/3.1.20.0/dev_guide/index.html>**

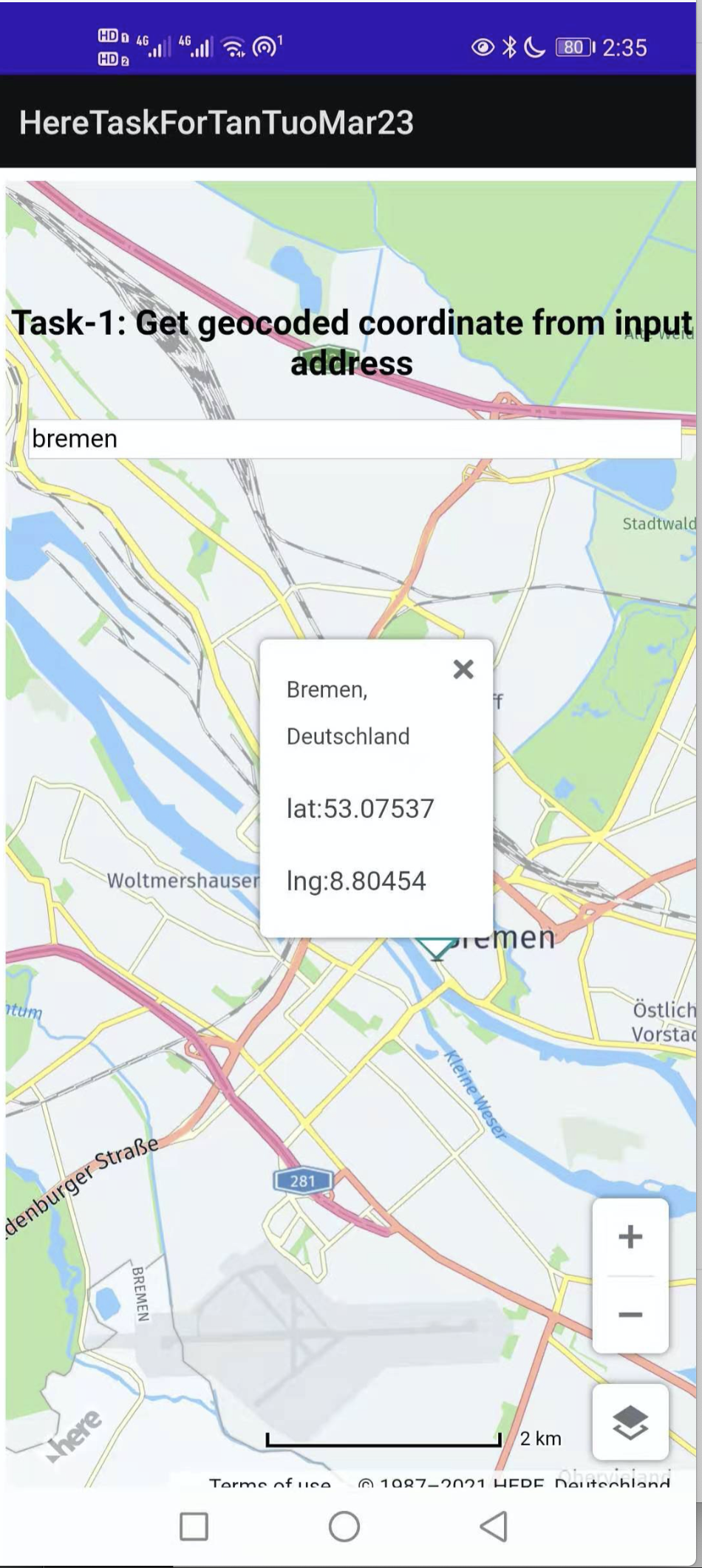
Answer :

I applied a Here API-KEY, implemented the application with HERE Javascript API, and show correct geocoded coordinate on the map based on an address input.

Please check following screen-shot from Android Mobile Device, I used webView to display Here JavaScript Map, when user put address into input box ( like Bremen, Munich for example ), Here map will show correct geocoded coordinate.

I also put JavaScript code demonstrated in following.

For completed Android full code, you can check it on my GitHub repository: *https://github.com/18601949127/HereTask01-JavaScriptAPI*



JavaScript key codes ( completed codes in *https://github.com/18601949127/HereTask01-JavaScriptAPI* ):

<!DOCTYPE html>  
<html>  
<head>  
 <meta name="viewport" content="initial-scale=1.0, width=device-width"/>  
 <link rel="stylesheet" type="text/css"  
 href="https://js.api.here.com/v3/3.1/mapsjs-ui.css?dp-version=1578490236"/>  
 <script type="text/javascript" src="https://js.api.here.com/v3/3.1/mapsjs-core.js"></script>  
 <script type="text/javascript" src="https://js.api.here.com/v3/3.1/mapsjs-service.js"></script>  
 <script type="text/javascript" src="https://js.api.here.com/v3/3.1/mapsjs-ui.js"></script>  
 <script type="text/javascript"  
 src="https://js.api.here.com/v3/3.1/mapsjs-mapevents.js"></script>  
  
</head>  
<body>  
<div id="test" style="width: 100%; height: auto; position: fixed; top: 10%;  
 left: 0; z-index: 99; text-align: center;">  
 <strong style="font-size: large;">Task-1: Get geocoded coordinate from input address</strong></br>  
 <br/><input type="text" id="auto-complete"  
 style="margin-left:5%; margin-right:5%; min-width:90%"  
 onkeyup="return autoCompleteListener(this, event);">  
</div>  
<div id="map" style="position:absolute; width:100%; height:100%; background:grey">  
</div>  
<div id="panel"  
 style="position:absolute; width:49%; left:51%; height:100%; background:inherit; display: none;"></div>  
  
<script type="text/javascript" charset="UTF-8">  
  
var AUTOCOMPLETION\_URL = 'https://autocomplete.geocoder.ls.hereapi.com/6.2/suggest.json',  
 ajaxRequest = new XMLHttpRequest(),  
 query = '';  
  
function autoCompleteListener(textBox, event) {  
  
 if (query != textBox.value){  
 if (textBox.value.length >= 1){  
  
 var params = '?' +  
 'query=' + encodeURIComponent(textBox.value) +  
 '&beginHighlight=' + encodeURIComponent('<mark>') +  
 '&endHighlight=' + encodeURIComponent('</mark>') +  
 '&maxresults=5' + // The upper limit the for number of suggestions to be included  
 '&apikey=' + APIKEY;  
 ajaxRequest.open('GET', AUTOCOMPLETION\_URL + params );  
 ajaxRequest.send();  
 }  
 }  
 query = textBox.value;  
}  
  
  
  
function onAutoCompleteSuccess() {  
 clearOldSuggestions();  
 addSuggestionsToPanel(this.response); // In this context, 'this' means the XMLHttpRequest itself.  
 addSuggestionsToMap(this.response);  
}  
  
  
function onAutoCompleteFailed() {  
 alert('Ooops!');  
}  
  
// Attach the event listeners to the XMLHttpRequest object  
ajaxRequest.addEventListener("load", onAutoCompleteSuccess);  
ajaxRequest.addEventListener("error", onAutoCompleteFailed);  
ajaxRequest.responseType = "json";  
  
  
// set up containers for the map + panel  
var mapContainer = document.getElementById('map'),  
 suggestionsContainer = document.getElementById('panel');  
  
//Step 1: initialize communication with the platform  
var APIKEY = 'wEaRjHxz2fWQOSKryZGfNPo0xfwOywM6VN1v6plgB10';  
  
var platform = new H.service.Platform({  
 apikey: APIKEY,  
 useCIT: false,  
 useHTTPS: true  
});  
var defaultLayers = platform.createDefaultLayers();  
var geocoder = platform.getGeocodingService();  
var group = new H.map.Group();  
  
group.addEventListener('tap', function (evt) {  
 map.setCenter(evt.target.getGeometry());  
 openBubble(  
 evt.target.getGeometry(), evt.target.getData());  
}, false);  
  
  
//Step 2: initialize a map - this map is centered over Europe  
var map = new H.Map(mapContainer,  
 defaultLayers.vector.normal.map,{  
 center: {lat:52.5160, lng:13.3779},  
 zoom: 3  
 });  
  
map.addObject(group);  
  
  
var behavior = new H.mapevents.Behavior(new H.mapevents.MapEvents(map));  
  
// Create the default UI components  
var ui = H.ui.UI.createDefault(map, defaultLayers);  
  
// Hold a reference to any infobubble opened  
var bubble;  
  
  
function openBubble(position, text){  
 if(!bubble){  
 bubble = new H.ui.InfoBubble(  
 position,  
 // The FO property holds the province name.  
 {content: '<small>' + text+ '</small>'+'<p>lat:'+position.lat+'</p>'+'<p>lng:'+position.lng+'</p>'});  
 ui.addBubble(bubble);  
 } else {  
 bubble.setPosition(position);  
 bubble.setContent('<small>' + text+ '</small>'+'<p>lat:'+position.lat+'</p>'+'<p>lng:'+position.lng+'</p>');  
 bubble.open();  
 }  
}  
  
  
  
function addSuggestionsToMap(response){  
 var onGeocodeSuccess = function (result) {  
 var marker,  
 locations = result.Response.View[0].Result,  
 i;  
  
 // Add a marker for each location found  
 for (i = 0; i < locations.length; i++) {  
 console.log(8888,locations[i].Location.DisplayPosition.Latitude,locations[i].Location.DisplayPosition.Longitude)  
 marker = new H.map.Marker({  
 lat : locations[i].Location.DisplayPosition.Latitude,  
 lng : locations[i].Location.DisplayPosition.Longitude  
 });  
 marker.setData(locations[i].Location.Address.Label);  
 group.addObject(marker);  
 }  
  
 map.getViewModel().setLookAtData({  
 bounds: group.getBoundingBox()  
 });  
 if(group.getObjects().length < 2){  
 map.setZoom(15);  
 }  
 },  
  
 onGeocodeError = function (error) {  
 alert('Ooops!');  
 },  
  
 geocodeByLocationId = function (locationId) {  
 geocodingParameters = {  
 locationId : locationId  
 };  
  
 geocoder.geocode(  
 geocodingParameters,  
 onGeocodeSuccess,  
 onGeocodeError  
 );  
 }  
  
  
  
 response.suggestions.forEach(function (item, index, array) {  
 geocodeByLocationId(item.locationId);  
 });  
}  
  
  
  
function clearOldSuggestions(){  
 group.removeAll ();  
 if(bubble){  
 bubble.close();  
 }  
}  
  
  
function addSuggestionsToPanel(response){  
 var suggestions = document.getElementById('suggestions');  
 suggestions.innerHTML = JSON.stringify(response, null, ' ');  
}  
  
  
  
var content = '';  
content += '<br/><strong>Response:</strong><br/>';  
content += '<div style="margin-left:5%; margin-right:5%;"><pre style="max-height:235px"><code id="suggestions" style="font-size: small;">' +'{}' + '</code></pre></div>';  
  
  
suggestionsContainer.innerHTML = content;  
  
  
</script>  
</body>  
</html>

**Task 2:**

**Please create a java program to find shortest path from A to F below Graph.**

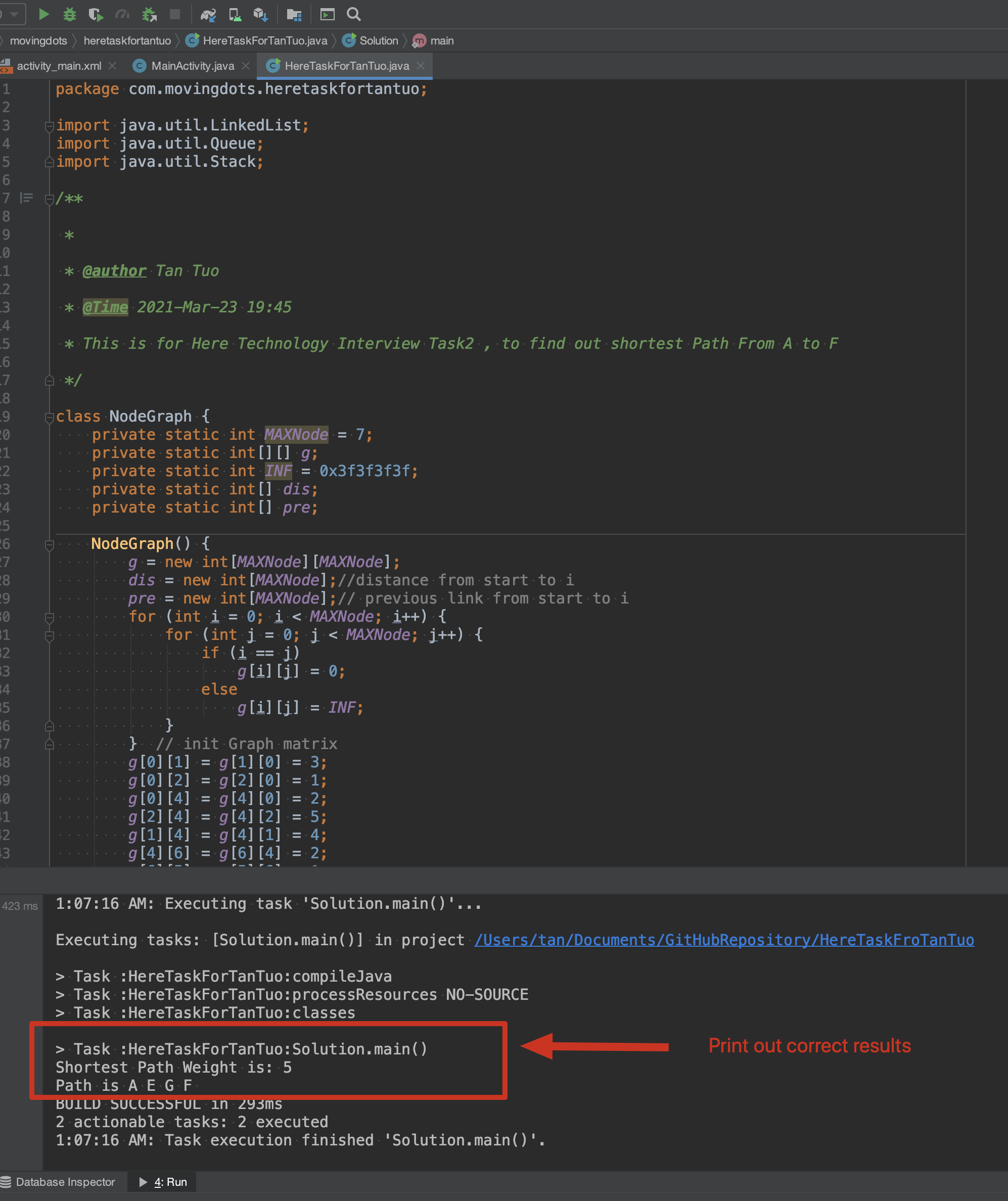
Diagram

Description automatically generated

Answer:

Please check with my program result in following IDE screen-shot , red square highlight the correct result of shortest path From A to F.

I used Dijkstra algorithm to get this result, and also attached Java codes in following :



JAVA Codes:

package com.movingdots.heretaskfortantuo;

import java.util.LinkedList;

import java.util.Queue;

import java.util.Stack;

/\*\*

\* @author Tan Tuo

\* @Time 2021-Mar-23 19:45

\* <p>

\* This is for Here Technology Interview Task2 , to find out shortest Path From A to F

\*/

class NodeGraph {

private static int MAXNode = 7;

private static int[][] g;

private static int INF = 0x3f3f3f3f;

private static int[] dis;

private static int[] pre;

NodeGraph() {

g = new int[MAXNode][MAXNode];

dis = new int[MAXNode];//distance from start to i

pre = new int[MAXNode];// previous link from start to i

for (int i = 0; i < MAXNode; i++) {

for (int j = 0; j < MAXNode; j++) {

if (i == j)

g[i][j] = 0;

else

g[i][j] = INF;

}

} // init Graph matrix

g[0][1] = g[1][0] = 3;

g[0][2] = g[2][0] = 1;

g[0][4] = g[4][0] = 2;

g[2][4] = g[4][2] = 5;

g[1][4] = g[4][1] = 4;

g[4][6] = g[6][4] = 2;

g[6][5] = g[5][6] = 1;

g[1][5] = g[5][1] = 3;

g[1][3] = g[3][1] = 6;

g[3][5] = g[5][3] = 2;

}

public void getShortestPath(int start, int last) {

for (int i = 0; i < MAXNode; i++)

dis[i] = INF;

for (int i = 0; i < MAXNode; i++)

pre[i] = -1;

dis[start] = 0;

Queue<Integer> q = new LinkedList<>();

q.add(start);

while (!q.isEmpty()) {

int u = q.poll();

for (int i = 0; i < MAXNode; i++) {

if (dis[i] > dis[u] + g[u][i]) {

dis[i] = dis[u] + g[u][i]; // update shortest distance to i

pre[i] = u;

q.add(i);

}

}

}

printShortestPath(last);

}

private static void printShortestPath(int last) {

Stack<Integer> st = new Stack<>();

System.out.print("Shortest Path Weight is: " + dis[last] + "\n");

while (last != -1) {

st.add(last);

last = pre[last];

}

System.out.print("Path is ");

while (!st.empty()) {

System.out.print((char) ('A' + st.pop()) + " ");

} // print out stack

}

}

class ShortestPath {

public static void main(String[] args) {

NodeGraph graph = new NodeGraph();

graph.getShortestPath(0, 5); // From A to F

}

}