# 关键词搜索

## 加载地图

代码如下：

//创建地图实例   
var map = new BMapGL.Map("container");   
//设置中心点   
var centerPoint = new BMapGL.Point(117.151767, 34.221332);   
//地图初始化，同时设置地图展示级别   
map.centerAndZoom(centerPoint, 16);   
//开启鼠标滚轮缩放   
map.enableScrollWheelZoom(true);

界面如下：



以中国矿业大学南湖校区为中心

## 搜索超市/博学楼

使用 searchNearby 函数根据中心点、半径与检索词发起周边检索。代码如下：

// 本地关键词搜索   
function searchKey(){   
 // 清空之前所有搜索记录   
 map.clearOverlays();   
 // 清空路径规划结果   
 var result = document.getElementById("routeResult");   
 if (result!=null)   
 result.innerHTML = '';   
 // 获取关键词   
 var keyWord = document.getElementById("search").value;   
 var options = {   
 renderOptions: {   
 map: map,   
 autoViewport: true,   
 panel: "searchResult"   
 }   
 }   
 var local = new BMapGL.LocalSearch(map,options);   
 local.searchNearby(keyWord, centerPoint, 900);   
}

界面如下：



搜索超市结果

## 路径规划

用户输入起点和终点，选择出行模式为步行，骑行或驾车（校内无公交）。选择规划方式为自动规划时，先进行搜索，把返回的第一个POI结果作为路线的起点和终点，输出结果；选择选点规划时，用户在本城市内选择合适的起点和终点，然后输出结果。结果列表将容器中进行展示，对驾车路线规划无效。

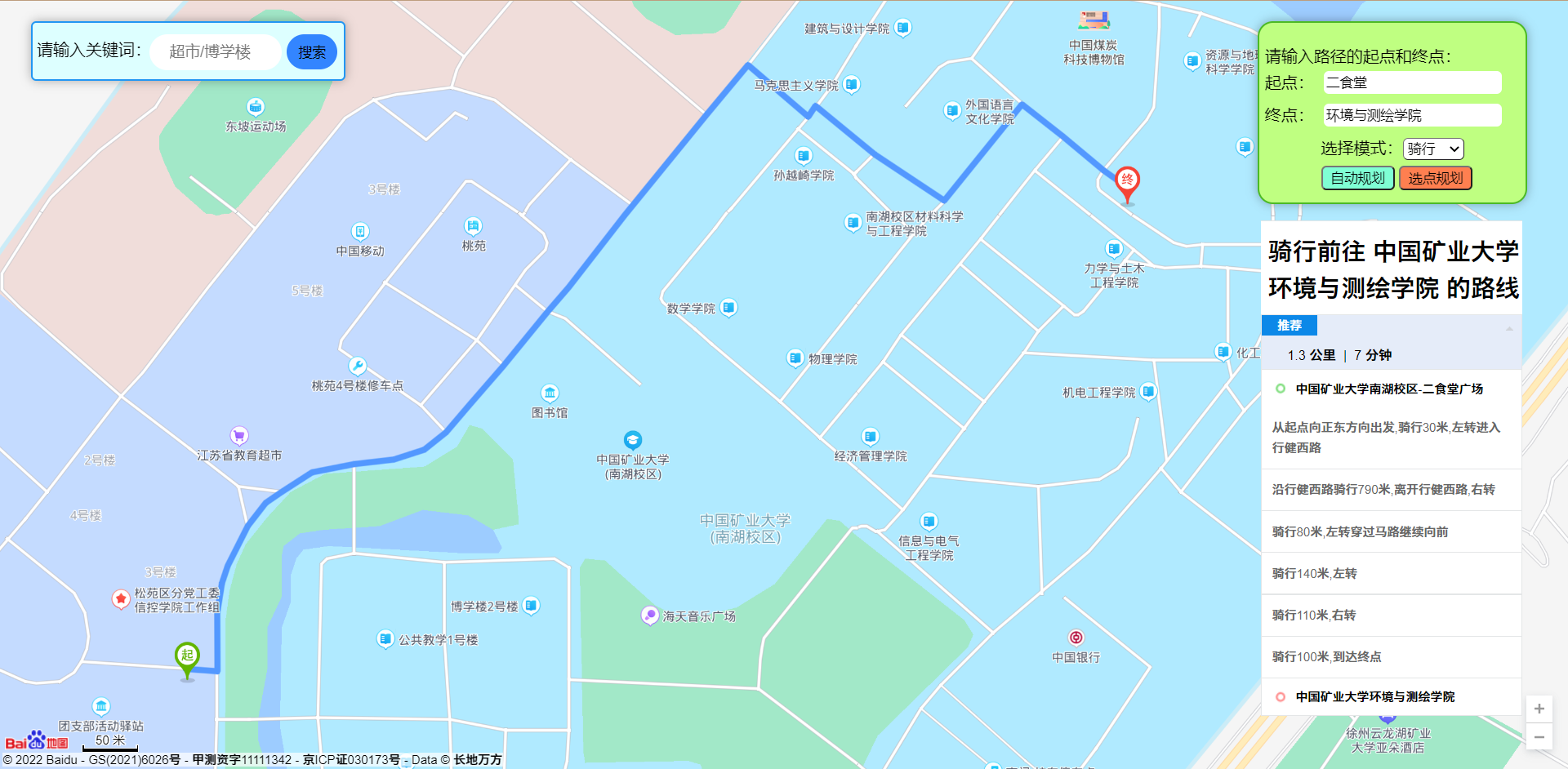
主要代码如下：

function autoNevigation(){   
 // 获取POI   
 // 清空之前所有搜索记录   
 map.clearOverlays();   
 // 获取关键词   
 var startKeyWord = document.getElementById("start").value;   
 var endKeyWord = document.getElementById("end").value;   
   
 var startPOI;   
 var endPOI;   
   
 var startOptions = {   
 onSearchComplete: function (results) {   
 // 判断状态是否正确   
 if (startLocal.getStatus() == BMAP\_STATUS\_SUCCESS) {   
 // 返回第一个搜索结果作为路径规划的起点   
 startPOI = results.getPoi(0);   
 console.log(startPOI);   
 var endOptions = {   
 onSearchComplete: function (results) {   
 // 判断状态是否正确   
 if (endLocal.getStatus() == BMAP\_STATUS\_SUCCESS) {   
 // 返回第一个搜索结果作为路径规划的终点   
 endPOI = results.getPoi(0);   
 // console.log(endPOI);   
 var select = document.getElementById("routeSelect").value;   
 if (select == 'walking')   
 walkingRouteSearch(startPOI,endPOI);   
 else if (select == 'riding')   
 ridingRouteSearch(startPOI, endPOI);   
 else   
 drivingRouteSearch(startPOI,endPOI);   
 }   
 }   
 }   
 var endLocal = new BMapGL.LocalSearch(map,endOptions);   
 endLocal.searchNearby(endKeyWord, centerPoint, 1000);   
 }   
 }   
 }   
 // 本地半径查询   
 var startLocal = new BMapGL.LocalSearch(map,startOptions);   
 startLocal.searchNearby(startKeyWord, centerPoint, 1000);   
   
 // 骑行规划   
 function ridingRouteSearch(startPOI, endPOI){   
 // 清空搜索结果   
 var result = document.getElementById("searchResult");   
 if (result!=null)   
 result.innerHTML = '';   
   
 var riding = new BMapGL.RidingRoute(map, {   
 renderOptions: {   
 map: map,   
 panel: "routeResult",   
 autoViewport: true   
 },   
 });   
 riding.search(startPOI, endPOI);   
 }   
 // 步行规划   
 function walkingRouteSearch(startPOI, endPOI){   
 // 清空搜索结果   
 var result = document.getElementById("searchResult");   
 if (result!=null)   
 result.innerHTML = '';   
   
 var walking = new BMapGL.WalkingRoute(map, {   
 renderOptions: {   
 map: map,   
 panel: "routeResult",   
 autoViewport: true   
 },   
 });   
 walking.search(startPOI, endPOI);   
 }   
 //   
 function drivingRouteSearch(startPOI, endPOI){   
 // 清空搜索结果   
 var result = document.getElementById("searchResult");   
 if (result!=null)   
 result.innerHTML = '';   
   
 var driving = new BMapGL.DrivingRoute(map, {   
 renderOptions:{   
 map: map,   
 panel: "routeResult",   
 autoViewport: true   
 }   
 });   
 driving.search(startPOI,endPOI);   
 }   
}

界面如下：



一食堂步行到达环测学院大楼



二食堂骑行到达环测学院大楼



三食堂驾车到达环测学院大楼

# 实验小结

前期尝试了天地图API，发现不够大众化，超市搜索结果不够全面；同时腾讯地图API封装做得不好，调用比较麻烦；最后选择了百度地图API。利用API开发应先把功能写好，再考虑美化设计。最后，开发路径规划功能时遇到了回调地狱问题，这也是调用第三方API的一个弊端。本程序最大的特点就是不限制路径规划的输入，三种出行模式和两种规划模式都可以输出结果。