//Functions list

//I have no time to make a beautiful table to store these Functions, sooooooorry 'bout that

//And I have no time to solve the problems of Github, soooooooooorry 'bout that

//confirm arguments and return value by yourselves

//@ 2019.05.11 Ur great group leader :/

//initialize module 初始化模块

//including building graph module and order module

void initial\_Coor\_G()//initial coordinate graph 初始化二维图

//直接用initial\_Coor\_G(CG)来初始化二维坐标图

void initial\_Rel\_G()//initial relational graph 初始化关系图

//直接用initial\_Coor\_G(CG)来初始化关系图

void initial\_Ord\_Q()//initial order queue 初始化订单队列

//initialize inner module

//these Functions should be only used in the Functions of initial module

//@ add new inner functions 2019.5.15 20:53 by Li

int judgeRow(int index)//私有函数，用于判断道路节点坐标图中道路节点的纵坐标

int judgeCol(int row ,int index)//私有函数，用于判断道路节点坐标图中道路节点的纵//坐标

void confirm\_Id()//confirm which node is road / cli / res 确定点的身份

int assign\_Ind()//assign index to every road\_nodes 给道路节点分配index

void connect\_Rel()//connect relaitionship between res / cli and road 联系res/cli与road

//input / output module 输入/输出模块

//Please using function freopen to redirect stdin and stdout into file

//then maybe we can use the functions of stdin and stdout instead of file to

//manage input and output

//we ignore output in animation for a while

//ATTENTION: If we use freopen, the input and output of file and terminal are the same.

//用freopen()

orderlistptr input\_File()//input by file 文件输入，返回链表头节点、更新//num\_of\_orders、结构体数组

orderlistptr input\_Ter()//终端输入，返回链表头节点、更新num\_of\_orders、结构体数//组

void output\_File()//output to file 文件输出，调用时（）内为两个输出链表的节点（两个节点）

void output\_Ter()//output to terminal 输出到终端，调用时（）内为两个输出链表的节点（两个节点）

[//@](mailto:/@2019.05.21)2019.05.21 Wu Chuanyv/

//duipai(in Chinese) module 对拍模块

//using random method to generate data ,which can be used to test programs

//U can ignore this module.

void data()

//simulate module 模拟模块

void floyd()//floyd to find to shortest path between road\_nodes floyd找最短路

void find\_Sta\_P()//find the start point 找起点

int SLP\_Ro()//get the shortest length of path between road\_nodes and restaurant\_nodes or client\_nodes (I change SPL into SLP)

//找道路节点和res/cli节点的SLP

//下同

int SLP\_Res\_Res()//get the shortest length of path between res\_nodes and res\_nodes

int SLP\_Res\_Cli()//get the shortest length of path between res\_nodes and cli\_nodes

int SLP\_Cli\_Cli()//get the shortest length of path between cli\_nodes and cli\_nodes

//the SLP Functions essentially are SLP\_Ro

//Remember to record the path using initial\_Tree() 记得记录路径在树中

//If U R free U can find which are LLP(longest length path as definition)

int get\_Tol\_Time()//get the total time of current rider 计算总时长

void get\_Inv\_Ord()//the orders the current rider cannot deliver in this bout（回合）, we call "invalid orders"

//这一轮当前rider送不了的订单

//we need to judge if the invalid orders can be delivered by the same rider in next bout

//we need this kind of judge again and again until the current rider cannot find any valid order in new about

void judge\_Step()//core algorithm ,which is to judge the next step of current rider using the tree

//获取路径 通过树

//then we can update

//update module 更新模块也是监控模块

//also monitor modle

void up\_Posi\_Rider()//update the position of rider 更新骑手位置——每个时间点更新时用循环对每个骑手都调用一次此函数——括号内：（第二个输出链表的节点、这个时间点骑手坐标x、y，骑手编号）

void up\_Num\_Rider()//If we need more riders, using up\_Num\_Rider 更新骑手数量——括号内：（上个时间点骑手总数量、第一个输出链表的节点）

void up\_Time\_Beg()//update the cost of time from beginning 更新时间——括号内：（上个时间点、第一个输出链表的节点）

void up\_Time\_Ord()//update the rest of time of order 更新订单还剩下的时间——（接单总数、数组t\_stop[ ]）

void up\_Visited\_Res()//update the set of visited\_Res 更新集合visited\_Res——（数组visit\_Res[][]、该节点位置坐标x、y,flag）

void up\_Visited\_Cli()//update the set of visited\_Cli 更新集合visited\_Cli——（数组visit\_Cli[][]、该节点位置坐标x、y,flag）

void up\_Money()//update the situation of "money" ,including bankrupt 更新钱的情况——（第一个输出链表的节点）

//I thought U'd need an argument of "flag" to choose the situation

//1. If we have visited this node(res / cli), we update the set

//2. If there is another same order, we alse need to update the set

//test module 测试模块

//some module we need to test can be seen in display\_up

//these all need to be displayed 这个模块的函数都要显示出来

//ATTENTIOM: display labels that we assign to various test Function 记得显示测试模块的时候写个label做标识

void test\_Rel\_G()

void test\_Coor\_G()

void test\_Tree()

void test\_ord\_Q()

void test\_SLP()//test all kinds of SLP using Rel\_G 显示各种SLP 用关系图

void display\_Up()//display update module 显示更新模块的数据

//用英文的原因

//1. 避免不同系统兼容性问题导致中文乱码显示

//2. 打字更少 写得更快

//3. 不用浪费力气和脑神经做中英文转换

//If U need any help or Function list need to be added something, let me know

//God bless team 28.

//@ 2019.05.11 Ur sad group leader :(