### 1 前提条件

#### 1.1 基础软件

电脑中装载有MongoDB及MongoDB Compass，并且设置好相关的环境变量Path。

#### 1.2 语言包

配备有Golang运行的环境，并且保证已经配备好了mgo包。如果没有配备好，在系统命令行中一次运行命令：

①go get gopkg.in/mgo.v2

②go get gopkg.in/mgo.v2/bson

### 2 概述

#### 2.1 名称：

银行固定资产盘查辅助系统

#### 2.2 功能：

##### 2.2.1 对MongoDB内的数据进行外部导入、额外添加、查找数据、修改数据和删除数据五种操作。

##### 2.2.2 从CSV文件中识别固定资产坐标，对MongoDB中对应数据进行修正。

##### 2.2.3 接受抽查百分比的输入，计算出盘查数量。

##### 2.2.4 随机确定对应数量的固定资产，显示出被抽中的固定资产名字。

##### 2.2.5 根据抽出的固定资产计算相互之间的直线距离，抽象出一个二维数组作为图。

##### 2.2.6 根据抽象图计算从每个固定资产位置出发的最短路径，用箭头连接显示路径。

### **3 注意事项**：

#### 3.1 基本顺序

##### 3.1.1 要保证第一遍使用时按照提示给出的序号运行一遍，然后才可以随意使用。因为要保证MongoDB中要有数据，否则容易崩溃。

#### 3.2 外部文件要求

##### 3.2.1 txt文件每一行的数据顺序按照名称、进账日期、规格型号、存放地点、价值写入，并且使用空格分开，每行最后一个数据之后一定要是回车，最后一行的最后一个不能有留空格

##### 3.2.2 csv文件每个名称之间用逗号隔开，如果某一格没有则不写，但是必须要有两个逗号，尾部的回车要求如上一条所示。

### 4 具体函数

#### 4.1 函数列表：

type FixedAsset struct

type CsvTable struct

type CsvRow struct

func MangoDBManage(c \*mgo.Collection)

func IntroduceInformation() <-chan FixedAsset

func AddInformation() <-chan FixedAsset

func SaveInformation(chanPerson <-chan FixedAsset, c \*mgo.Collection) bool

func FindInformation(c \*mgo.Collection)

func UpdataInformation(c \*mgo.Collection)

func RemoveInformation(c \*mgo.Collection)

func CsvIntroduce(c \*mgo.Collection)

func CsvRead(file \*os.File, row int, c \*mgo.Collection)

func RandomSelect(c \*mgo.Collection) []FixedAsset

func FloatToInt(f float64) int

func Random(extent int) int

func GenerateShortestPath(c \*mgo.Collection, randomasset []FixedAsset)

func minpath(v int, csvmap [20][20]int, a []FixedAsset)

#### 4.2 具体细节

##### 4.2.1 type FixedAsset struct

里面设有7个成员，对应固定资产的名称、进账日期、规格型号、储存地点、价值、所在CSV图横纵坐标。

##### 4.2.2 type CsvTable struct

设有1个成员，是type CsvRow struct的切片。与其连用刻画一个字符串的二维数组。

##### 4.2.3 type CsvRow struct

设有1个成员，是string的切片。它主要用于储存CSV中一行的字符串数据，用于刻画CSV表格的关系。

##### 4.2.4 func MangoDBManage(c \*mgo.Collection)

传递参数：

mgo包中Collection的指针类型，传入指向MongoDB对应集合位置的指针

返回参数：

无

作用：

1.像main函数一样通过输出提示获得用户的输入选择

2.调用从外部的文本文件直接导入大量固定资产数据信息，方便用户录入

3.在文本导入的基础上（如果有）查找指定数据并标准输出信息或者输出数据库全部信息

4.添加额外的数据信息

5.通过名字修改指定数据项的信息

6.删除指定数据项或者全部数据项（使用全部删除时会再次进行确认，可以反悔）

##### 4.2.5 func IntroduceInformation() <-chan FixedAsset

传递参数：

无

返回参数：

可以缓存FixedAsset结构类型数据的通道类型

作用：

MongoDBManage函数调用的外部导入函数，用于导入外部文本文件数据

##### 4.2.6 func AddInformation() <-chan FixedAsset

传递参数：

无

返回参数：

可以缓存FixedAsset结构类型数据的通道类型

作用：

MongoDBManage函数调用的添加数据函数，用于在MongoDB中额外添加数据

##### 4.2.7 func SaveInformation(chanPerson <-chan FixedAsset, c \*mgo.Collection) bool

传递参数：

可以缓存FixedAsset结构类型数据的通道类型，传入缓存有外部导入或输入的该结构通道

mgo包中Collection的指针类型，传入指向MongoDB对应集合位置的指针返回参数：

bool类型

作用：

MongoDBManage函数调用的信息储存函数，用于把导入和新添加的数据存入MongoDB中

##### 4.2.8 func FindInformation(c \*mgo.Collection)

传递参数：

mgo包中Collection的指针类型，传入指向MongoDB对应集合位置的指针返回参数：

无

作用：

MongoDBManage函数调用的查询函数，用于查询指定一项文件或者输出全部文件

##### 4.2.9 func UpdataInformation(c \*mgo.Collection)

传递参数：

mgo包中Collection的指针类型，传入指向MongoDB对应集合位置的指针返回参数：

无

作用：

MongoDBManage函数调用的修改函数，用于修改MongoDB中已经存在的数据

##### 4.2.10 func RemoveInformation(c \*mgo.Collection)

传递参数：

mgo包中Collection的指针类型，传入指向MongoDB对应集合位置的指针返回参数：

无

作用：

MongoDBManage函数调用的删除函数，用于删除指定一项或者所有数据

##### 4.2.11 func CsvIntroduce(c \*mgo.Collection)

传递参数：

mgo包中Collection的指针类型，传入指向MongoDB对应集合位置的指针返回参数：

无

作用：

找到并且读入CSV座位表，识别各数据所在位置

##### 4.2.12 func CsvRead(file \*os.File, row int, c \*mgo.Collection)

传递参数：

os包中的文件字段指针类型，是指向将要打开的文件的指针

整数类型，传入CSV中的行数

mgo包中Collection的指针类型，传入指向MongoDB对应集合位置的指针

返回参数：

无

作用：

根据识别的位置把MongoDB中存入的对应数据项的横纵坐标值更正

##### 4.2.13 func RandomSelect(c \*mgo.Collection) []FixedAsset

传入参数：

mgo包中Collection的指针类型，传入指向MongoDB对应集合位置的指针

返回参数：

FixedAsset类型的切片，返回储存随机抽查数据的切片

作用：

从MongoDB中调取所有的数据项作为备选，接受用户的比例输入计算抽取个数

调用Random生成一组随机数（各不相同）并以此作为盘查项的序号，取出对应序号的备选项

##### 4.2.14 func FloatToInt(f float64) int

传入参数：

浮点数，传入抽查的计算结果

返回参数：

整数，返回计算结果的近似整数

作用：

由于用户输入的百分数很有可能导致计算结果不是整数，所以要把计算结果转换为整数

##### 4.2.16 func Random(extent int) int

传入参数：

整数，在for循环中多次使用时最好传入变化的整数（不能为0）

返回参数：

整数，根据传入的整数得到的随机数

作用：

生成一组随机数

##### 4.2.17 func GenerateShortestPath(c \*mgo.Collection, randomasset []FixedAsset)

传入参数：

mgo包中Collection的指针类型，传入指向MongoDB对应集合位置的指针

FixedAsset类型的切片，传入保存有随机抽取的数据结果的切片

返回参数：

无

作用：

从MongoDB中获得所有盘查数据的坐标值，计算各点间的直线距离并生成抽象出来的二维数组图中

##### 4.2.18 func minpath(v int, csvmap [20][20]int, a []FixedAsset)

传入参数：

整数，传入计算最短路径的起点

二维数组，传入在GenerateShortestPath中抽象出的路径图

FixedAsset类型的切片，传入保存有随机抽取的数据结果的切片

返回参数：

无

作用：

寻找最短路径并输出，也就是prim最小生成树算法