整体的设计思路以及完成情况

1、原始数据的获取与处理：  
地图数据：  
在线获取地图的图片文件和标注信息XML文件，程序开始时调用初始化程序，将地图的标注信息文件读取到内存，存储房间位置等的地图信息，构建一个树。树的结构如下:每一个节点都有其包含的单元的信息，以及与之拓扑相连的单元的指针。  
房间和走廊有矩形和多边形两种形状。  
  
可以实现功能：1>提供绘制地图的数据，根据xml文件解析出地图中的房间等的位置信息以 及拓扑关系。  
 2>查询对应坐标的房间信息;

3>查询多个房间的拓扑关系;

主要是判断是否两个房间和走廊之间是否有门。  
 4>查询轨迹是否穿墙、坐标点是否在不可能区域，以及轨迹是否经过门。  
 5>获取WiFi热点的MAC地址和具体位置。  
传感器数据：  
定期获取传感器数据。并分别进行滤波处理原始的数据。  
 三轴加速度: （低通滤波）||(扩展卡尔曼滤波)  
 三轴磁阻:（低通滤波）  
 陀螺仪: （低通滤波）||(扩展卡尔曼滤波)  
WiFi热点信息：定期获取WiFi强度信息，利用最小二乘法解算出具体位置并估计方差。

2、定位流程  
确定初始位置：

由WiFi定位，结合地图信息确定。具体方法为，先由WiFi定位给出初始点以及估计方差。根据WiFi定位的信息，利用服从数据的二维正太分布随机生成N个粒子、权重为1/N。然后根据地图信息去掉一些粒子（如在不可到达位置），重新调整权重。  
正常定位：  
模态选择：根据传感器数据判断用户的状态（利用低通滤波处理传感器的数据分析数据基波的幅值和频率判断用户的状态）：  
行走：  
 手机放在包里或随臂摆动：周期恒定,约为0.5-1Hz;幅值较大  
 手机拿在手里发短信或是：周期恒定,约为0.5-1Hz;幅值较小  
 手机在打电话：周期恒定,约为0.5-1Hz;幅值较小  
站立（静止）：无恒定周期的基频信号  
行车：无恒定的基频信号（建议单独设置一个行车模式，由用户手动开启）

模态的定义分成两个层次：第一个层次是人的运动模式，可以分为静止、行走、跑步、行车四种状态；同时，手机有其自身的状态模式，如手机在手中texting，browsing；随着手臂swing摇摆、晃动；在口袋中；在提包、手袋中；正在放入口袋；正在从口袋取出来。这两个层次是可以组合的，比如行走中打电话、看短信或者微信（texting）、或者拿在手里晃来晃去。

上述情形在sensors-13-01539.pdf中有比较明确的描述，以及相应的算法可以参考。  
行走状态：  
步长、步频、步向的确定。其中步长初始由一个预设值（经验值），然后在后期的定位过程中根据定位信息修正步长，然后作为以后用户的步长（可以分别记录用户在不同步频下的步长）。步频可由对信号进行频域的分析得到。步向由磁场感应，和陀螺仪的数据得到。利用粒子滤波对航迹进行分析（用样本均值代替对概率密度函数的积分）。（另考虑部分手机传感器不全的情况，如没有加速度感应和陀螺，通过用其他传感器代替）

数据融合：根据处理后的传感器（三轴加速度、陀螺仪）数据进行步监测，若幅值超过一个参考值，则认为其已迈出一步，根据三轴磁阻和方向传感器以及陀螺仪得到当前的步向，以及步长信息调整粒子的位置信息和权重信息：  
  
根据调整后的粒子权重去掉权重小于1/N²的粒子，将权重大于2/N的粒子分裂.  
然后根据地图信息去掉一些粒子（如在不可到达位置，前后位置穿墙），重新调整权重。然后根据返回的WiFi定位信息调整当前轨迹和修正步长.

行车时:  
根据WiFi定位,确定初始位置，预设一个行车速度，并根据地图信息给出当前最有可能的行车方向。然后检测传感器数据，根据方向传感器，陀螺仪，加速度传感器等判断车辆的拐弯方向以及角度，然后在根据地图的信息调整行车速度。

## 目前我的部分完成情况：

地图：  
数据格式：最大的是floor类，其会包含一系列的房间、走廊和门以及他们的位置和拓扑关系的信息。其包含的功能为：返回一个坐标的房间（走廊），返回两个房间（走廊）的位置关系（有无门连接），判断一段轨迹是否穿墙，或者是经过门。  
房间和和走廊均支持矩形和多边形两种形态（封闭）；  
核心算法是判断一个点是否在一个多边形内部：利用在房间外的一点和当前位置连线判断和 所有边的交点（与顶点相交不算）个数，若为奇数则在多边形内部，若为偶数则在多边形外部。

模态检测：  
目前主要实现了步监测：计算在一个滑动窗口内窗口数据的均差，在连续计数到m个方差值大于T的点，以及n个方差小于T的点时认为走了一步。  
问题：在手持时会漏掉一些步。

步向判断算法：  
最开始采用的算法：利用两个周期内的三轴加速度的传感器数据，求平均值得到的加速度的向量ag应为重力加速度（即垂直于地面的方向），然后将将三轴磁阻m1和三轴加速度a1的半个周期期内的平均向量投影到与ag垂直的平面上去，计算a1’`和m1’的夹角，作为步向。

问题：理论上不受手机姿态的限制，但实际测试时误差太大。  
简单基于磁阻的算法：根据手机x，y轴的磁阻得到手机当前的方向作为步向。（）  
问题：在手持时精度可以，但无法处理其他情况。

粒子滤波算法：(主要参考06418907.pdf)

初始化：根据WiFi定位返回的定位信息（x，y）以及标准差（δx， δy），利用正太随机分布（x，y，δx， δy，0）产生数目为T的粒子。  
预测：  
根据返回的（L，D，δl，δd）更新粒子的位置。然后根据粒子现在的位置p（x，y）与前一时刻的位置pf（xf，yf）判断粒子的轨迹是否合理：  
若p与pf在同一房间（走廊）内：则不删除粒子；  
若p与pf不在同一房间（走廊），但两个房间（走廊）有门连接，且p与pf的连线的中点在一门所在直线为直径的圆内：则不删除粒子；  
其余情况：删除该粒子。  
然后根据所有存活粒子位置信息计算出其均值（x~,y~）以及方差（δx，δy），根据N（x~，y~，δx，δy，0），将每个粒子的位置带入函数表达式，更具返回结果调整权重。  
最后根据每个粒子的权重挑中该粒子：  
若粒子权重小于1/T²，则删除改粒子。  
若粒子权重大于2/T，则将改粒子拆成2个相同的粒子。

归一化处理；

以上是目前实现的功能：其中粒子的动态生成还有一些问题。另还没有考虑如何利用WiFi位置信息矫正。

整体运行：（步长，步向的方差均先制定）  
初始化：  
先根据返回的地图数据进行xml 文件解析；初始化WiFi定位信息；待获取了当前WiFi热点的强度信息后，进行一次WiFi定位，然后根据返回的定位信息进行粒子滤波的初始化。  
定位：  
更具

附件说明：

地图部分的代码：  
rect.h----------------矩形类  
lines.h---------------多边形类  
Circe.h---------------圆类  
door.h---------------门类  
room.h--------------房间类  
passageway.h------走廊类  
floor.h----------------楼层类

粒子滤波部分的代码:  
particle.h-----------粒子类  
particle\_chain-----粒子链表类  
particle\_filter------粒子滤波类

传感器数据处理代码:  
sensor\_data.h----传感器数据类

地图解析部分代码  
tinystr.h-------------更改后的xml文件解析库  
tinyxml.h------------更改后的xml文件解析库  
read\_xml.h---------解析地图文件,初始化楼层信息,读取WiFi位置和ap信息