**Java&J2EE应用与开发**

学号： 201692077

姓名： 李云皓

大作业得分点

文档质量，程序注释，

编码规范及易读性（10分）：

功能实现（10分）：

异常处理（10分）：

类设计，线程同步（10分）：

大作业成绩（满分40分）：

平时作业成绩（满分60分）：

期末总成绩（满分100分）：

**消息传递机制**

**1.1 消息传递与解析机制**

消息与内容之间用空格隔开以利于消息的Responder进行解析。例如上线消息是”Online A”当Responder收到该消息时就会知道节点名称为A的点已经上线。

**1.2 消息类型**

|  |  |  |
| --- | --- | --- |
| 消息头 | 内容 | 备注 |
| Online | 节点名 | 结点上线时发送的消息 |
| Ack | 节点名 | 新上线节点能够通过该消息知道哪些节点依然在线 |
| Offline | 节点名 | 离线消息 |
| Check | Null | 检查邻居是非在线 |
| Response | 节点名 | 邻居返回的消息类型 |
| ShutDown | 节点名 | 发送邻居的掉线消息 |
| Request | 路径+空格+cost | 发送已经学习到的路径 |
| NeibourNode | 路径+空格+JSON | 回应Request请求 |

表1-2消息类型

**静态成员**

**2.1静态数据**

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 备注 |
| port | int | 端口号 |
| Servicer | DatagramSocket | 代表节点与互联网交换信息的Socket |
| name | String | 节点名 |
| RunningSteady | boolean | 表示节点已经稳定运行 |
| Configs | ArrayList | Config文件中读取的信息 |
| Neibours | ArrayList | 一跳之内能到达的在线的节点的相关信息 |
| Reach | ArrayList | 经过若干跳能够到达的节点的相关信息（包含路径） |
| ShortestPaths | ArrayList | 到达点的最短路径(包含路径) |
| ResponseStaticNodes | HashMap | 备用变量，在Heart类中使用。 |

表2-1 静态数据

StaticNode类内的数据和构造函数：

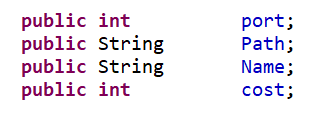
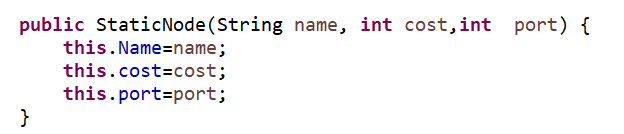
 

图2-1 StaticNode类内的数据和构造函数

**2.2静态函数**

**2.2.1字符串为通信的基本单位**





第一个函数会将Head和Body中间加一个空格拼到一起。并且将数据转换为一个可以经过Socket传递到目的端口号的数据包。到达目的地后，可以使用第二个函数进行解包。返回Kind+” ”+Messege字符串。使用以下函数可以方便解析数据：





在我们的程序中，经常要将形如”str1 str2 str3”这样的按照空格隔开的字符串进行分离。取名叫空格串





第一个函数用于获取空格串中包含的子串的个数

第二个函数用于获取空格串s中 第index号串

**2.2.2 类和字符串的相互转化**





以上两个函数会将StaticNode类型的数据

**类实现**

**2.1 Responder类**：

Responder类继承了Thread，用于无限接收和解析来自各个节点的消息。

收到一个数据包后Responder会做相应的操作，提取出DatagramPacket数据包中的内容。data=Node.*getPacketContent*(dp);

head=Node.*getHead*(data);

body=Node.*getBody*(data);

根据head的值，Responder会将消息进行分类。

**if**(head.equals("Offline")) {

**continue**;

}

类似的并列if就不一一列举。在各个发出请求的类中将详细介绍。

**2.2 Djikstra类：**

涉及数据类型：StaticNode类。

涉及消息类型：Request、NodeNeibour。

涉及静态变量：Neibours(ArrayList)，ShortestPath(ArrayList)，Reach(ArrayList)