RAT程序说明文档

1. UML类图
2. 设计思路

使用Dictionary，Word，Entry三个层次作为词典的基础。其中entry继承出很多不同了实现类，因为不同词性的义项包含的内容是不同的。其中Dictionary使用单件模式，仅在程序开始时自动构造一次，并调用read方法来读取资源文件中的内容，之后不能再次被构造。而Word，Entry的创建使用工厂模式，因为创建的过程十分复杂而且要判断Entry具体的实现类类型。

软件支持多个用户，每个用户拥有自己的set，这是一些属于Dictionary的单词的集合，用户的test和recite功能都是针对set而不是全部的单词（也可以把全部单词作为一个set）。Class User表示用户。Class set 表示集合。User中有vector<set\*>的属性保存用户的全部set。

Class interface表示界面。实现类consoleInterface表示命令行界面。用户的命令将在某些方法中进行分析然后传递给consoleInterface的其他方法进行执行。这个类承担所有和用户交互的工作。

对于测试功能，是有一个Question的基类。然后派生出了三个派生类，分别对应三种测试方法：根据单词选择意思（不定项选择，少选，错选均不对，下同）、根据意思选择单词（不定项选择）、根据意思拼写单词。通过更多种的测试方法，增强用户对指定set的了解。Question类的测试结果不仅会实时的反馈给用户，也会存储在相应单词的Test类中。Test类中存有相应单词的Entry在三种类型的测试下的正确的次数、错误的次数和正确率等。Test类中还存有此用户在整个使用期间的正确率（静态变量），会在每次测试后动态变化并返回给用户。所有的这些正确次数等都会在程序退出时存储在文件中，运行时再读进来。Test拥有一个根据正确率和测试次数给每个单词赋权值的机制，在随机选择测试的单词时，权值大的单词会有更大的概率被选进来。权值在每次测试后会进行更新，测试的次数越少，错误次数越多，单词的权值越大。

对于背诵功能，主要是由recite类来完成的。首先用户在创建set时需要输入一个天数，为自己计划背完这个set的天数，若天数不足后期也可以进行修改。recite类会记录开始背诵的日期和剩余的天数来安排每日的背诵量，并在进入程序的时候对用户进行提醒。我们参考了当今市场上比较好的背单词软件，安排每天的单词复习量为背诵量的2倍，达到较好的背诵效果。背诵时由ReciteControl函数来控制，DoRecite和DoReview函数用来背诵和复习某个特定的单词。开始背诵时，会根据当日的计划完成情况，从未背的单词中随机选取要背诵的单词，然后选取相应数量的需要复习的单词，加入背诵和复习计划。其中选取复习的单词时，会根据艾宾浩斯遗忘规律赋给每个单词权重，然后按照权重排序选取相应的单词，来达到最好的复习效果。在实际背诵中，根据已有的经验，会把12个单词左右作为一个小组，其中前几个为背诵单词，后面为复习单词。对于每个单词，都会记录每次背诵的时间，和正确的情况，对于背诵时间不同的单词，我们对其要求不同。只有在当日连续相应的次数Review的测试通过了，当日这个单词才算成功复习，退出当日复习列表。这样可以使新单词复习更多次，老单词加深印象，符合记忆遗忘规律。如果在Recite或Review的过程中发现某个单词特别熟悉，可以选择斩了这个单词，以后不会再出现在复习列表里。同样，如果某个单词累计复习成功次数达到8次，自动列为已经完全背会，也进入斩的列表。如果当日用户的背诵量完成了，可以选择额外增加背诵复习量。

1. 设计说明
2. Dictionary

属性包括Dictionary的一个实例（static）以及vector<Word>。

方法包括进行精确查询，匹配查询，子串匹配（kmp算法），插入，排序，读取文件等。

1. Word

属性有单词的拼写，vector<entry\*>,背诵的有关参数。

提供的方法包括排序，查找，判断一个单词是否熟悉等。重载了向屏幕输出的<<。

1. Entry

根据不同的词性有不同的属性。比如名词会记录是否是复数，是否是缩写，复数形式等。

重载了向屏幕输出的<<运算符，提供向资源文件输出的方法。

1. Factory

包括WordFactory的EntryFactory（抽象类）。主要承担分析资源文件并且生成product的过程。

1. Interface

属性包括用户，当前用户，以及一些运行参数（比如到目前是否有数据被修改）。

提供众多的方法：初始化，接受用户命令，分析用户命令，查询，测试，背诵......大部分通过对命令的分析调用其他类中的方法。用户登录，切换用户等是直接在这里实现了。

1. User

属性包括每个用户的用户名，密码，等级，vector<set\*>等。构造函数将从文件中读取用户的数据。

1. Set

包含vector<Word\*>。

方法包括查找，插入等。

1. Question

主要分为按照权重随机选择相应的问题(AddQuestion)、将问题随机打乱(DisorganizeQuestion)、将问题输出给用户(ShowInScreen)、获得用户的答案并给出判断解答(GetTheAnswer)四个虚函数。Operation类对其进行操作控制。

1. Test

主要包括相应的Entry的正确、错误的次数和正确率信息，也存有用户在整个的测试中的正确次数，错误次数，正确率的信息（静态变量）。

1. Recite

完成Recite和Review功能，能够进行背诵和复习，ReciteControl函数控制整个的背诵过程能够按照一定的遗忘规律进行。Recite中还含有一些计算相应权值的函数。