一、步骤：选择头文件（template.h），输入要生成的类名A。会在主目录下data生成A.h,A.cpp,A.hxx，头文件路径保存在主目录的config.txt

二、说明：

1. 会自动生成Initialize，SetFromPB，CreateTlvPB及其嵌套类的SetFromPB，CreateTlvPB，不管存不存在。
2. 关键字：

(1)嵌套类的命名，是假设C开头的，比如CA，其关键字为A，

(2)父类和生成的类，其关键字分情况：如果类似CTemplateA，关键字为A，CTplA，关键字为A，其他情况是去掉第一个字符剩下的为关键字。

(3)类的成员，假设是mA的形式，关键字为A。

关键字用于PB结构和函数的命名，比如类成员的PB命名一般是PB+关键字，嵌套类的Set命名为Set+关键字+FromPB(类名 & r+关键字，PB+关键字 \* pbData)

1. 非嵌套类，会假设该类已经有SetFromPB，CreateTlvPB，其形式为Set+关键字+FromPB(bool bServer,PB+关键字 \* pbData)
2. 对于二维数组支持，
3. 如果该成员是int，PB文件相应生成的字段是repeted PB+关键字 关键字 = N的形式，其PB+关键字 里面的内容为 repeated uint32 关键字 = 1
4. 如果该成员是string，PB文件相应生成的字段是repeted string 关键字 = N的形式
5. 如果该成员是类，PB文件相应生成的字段是repeted PB+关键字+Data 关键字+Data = N的形式，其PB+关键字+Data内容为repeted PB+关键字 关键字 = 1

二维以上的数组暂时不支持。

1. 对于一般情况，都能生成类的PB，比如property.h中的 CBaseProperty

三、不足：

1.调试不足，难免有小BUG，遇到后希望能反馈出 文件名 + 类名的信息，好修正（BUG还是好复现的。。。）

2.忽略枚举的定义以及假设类的成员为int，string，class，其后可考虑扩展（没啥大问题）

3.Create和Set的代码，写的冗余，很大优化空间

4.对于使用者，有假设的条件需要满足，如非嵌套类的set和create必须存在，其类的命名必须遵循主流命名。

1. 扩展：
2. 二维数组以上的支持
3. 对于不足的第4点，尽可能减少，方便使用
4. 对不足的第3点优化
5. 与其他工具的结合？