**本目录为dCHAREs相关工具：**

**chare.py为dCHAREs的文法构建程序，构建dCHAREs的文法和简洁文法**

**writingdCHAREs.py为dCHAREs的辅助书写的主程序，依据禁用终结符集交互书写dCHAREs**

**generator.py为dCHAREs的随机生成程序，输入字母表和表达式长度N，随机生成一个长为N的dCHARE**

**newdchare.py为dCHAREs的简洁文法的构建程序**

**newgenerator.py为优化后的dCHAREs的随机生成程序，基于dCHAREs的简洁文法实现**

**experiments.py为对生成dCHAREs程序的实验，统计不同字母表和表达式长度下的平均生成时间**

1. 运行环境

程序均由python语言开发，请先在本机安装python 3及其编译器，在编译器中打开“\*\*\*.py”等脚本文件即可运行和调试。

二、使用说明

* **chare.py**：

input:字母表大小

output：dCHAREs文法产生式，和文法产生式的数目

字母表用a,b,c...表示(预先定义字母表候选list,)

文法产生式有两种：

1：Fac(A->a)

X(R,F,α,β)->(S,α,β)

输出5元组：R,F,α,β,S

2：Seq(A->Ba)

X(R,F,α,β)->X(R1,F1,α1,β1)·(S,α,β)

输出9元组：R,F,α,β,R1,F1,α1,β1,S

* **writingdCHAREs.py：**

交互式辅助书写dCHAREs，提示当前禁止词表

字母表用a,b,c...表示(预先定义字母表候选list,)

文法产生式有两种：

1：Fac(A->a)

X(R,F,α,β)->(S,α,β)

2：Seq(A->Ba)

X(R,F,α,β)->X(R1,F1,α1,β1)·(S,α,β)

例：生成的句子为 (a),(b),(c),(d)

选取的产生式为

X3->a, X2->X3b, X1->X2c, X0->X1d

存储X0X1X2X3

静态显示：字母表全集S，操作符表全集O

用户输入：当前链式因子操作符ai和bi、终极符集合Si{c1,c2…ck}

提示：下一链式因子的操作符ai+1和bi+1与禁止词表Si+1的组合

若ai=bi=+，则当ai+1=bi+1=+，Si+1为空集

若ai或bi=\*或?，则当ai+1=bi+1=\*或？，Si+1=S-Si

* **generator.py：**

input:句子长度N

output：长度（终结符个数）为N的一个dchare

字母表用a,b,c...表示(预先定义字母表候选list,)

文法产生式有两种：

1：Fac(A->a)

X(R,F,α,β)->(S,α,β)

2：Seq(A->Ba)

X(R,F,α,β)->X(R1,F1,α1,β1)·(S,α,β)

* **newdchare.py**

构建dCHAREs的简洁文法

input:字母表大小

output：dCHAREs文法产生式和产生式数目

字母表用a,b,c...表示(预先定义字母表候选list,)

文法产生式有两种：

1：Fac(A->a)

X(R,F,t)->(S,t)

输出4元组：R,F,t,S

2：Seq(A->Ba)

X(R,F,t)->X(R1,F1,t1)·(S,t)

输出7元组：R,F,t,R1,F1,t1,S

* **newgenerator.py**

基于dCHAREs的简洁文法的生成

input:句子长度N

output：长度（终结符个数）为N的一个dchare

字母表用a,b,c...表示(预先定义字母表候选list,)

文法规则有两种：

1：Fac(A->a)

X(R,F,t)->(S,t)

2：Seq(A->Ba)

X(R,F,t)->X(R1,F1,t1)·(S,t)

例：生成的句子为 (a),(b),(c),(d)

选取的产生式为 X3->a, X2->X3b, X1->X2c, X0->X1d

存储X0X1X2X3