

# 编译原理——大作业语法分析

## 基本架构

lexya\_a.y 文件

lexya\_a.tab.h 文件

lexya\_a.tab.c 文件

语法规则.pdf

make 命令行脚本

parser 命令行脚本

clean 命令行脚本

## 执行步骤

运行环境： MacOS（如果想在Linux上运行则需要修改node.c的#include<sys/malloc.h>修改了#include<malloc.h>,然后直接在项目中make就能运行了）

- `./make` 编译  
    执行 `make` 前  
    执行 `make` 后
- `./parser test` 测试test文件  
    执行 `parse` 后的输出
- `./clean` 删除编译出的无关文件

## 输出结果

最终结果会显示在terminal以及写入out文件中。

注：若匹配出错，则out文件会显示error。同时产生一个stderr文件，用来说明第一个不匹配出现在代码的哪一行。

## 其他

在测试语法分析器时，有两个测试文件

1. `test` 文件，用来测试语法分析功能
2. `errTest` 文件，用来测试相应的报错机制。