

西北工业大学

《编译原理》语法分析实验

学 院： 软件学院

学 号： 2018303081

姓 名： 马泽红

专 业： 软件工程

西北工业大学

2021 年 4 月

目录

1.代码介绍.....	3
1.1 代码目录介绍.....	3
1.2 代码运行方式.....	3
2.自定义测试方案.....	4
2.1 struct_type 测试.....	4
2.2 数据类型测试.....	6
2.3 其他表达式综合测试.....	7
2.4 错误测试.....	8

1.代码介绍

此次语法分析实验的代码使用**递归下降算法**实现，在各匹配函数中完成各语法规则的匹配与抽象语法树的构建。

1.1 代码目录介绍

MIDL_compiler

```
└─src //程序源代码
    │
    └─inputFile //输入的测试文件
        │
        └─语法测试文件
            │
            └─base_type_spec
            │
            └─declarators
            │
            └─expr
            │
            └─member_list
            │
            └─struct_type
            │
            └─type_spec
            │
            └─自定义测试文件 //自己的测试方案存储的位置
        │
        └─CompilerTest.java //程序入口，输入源程序，输出语法分析结果
    │
    └─outputFile
        │
        └─syntaxAnalysisOutput //语法分析结果存储位置
        │
        └─tokenAnalysisOutput //词法分析结果存储位置
    │
    └─syntaxAnalysis //语法分析的源代码包
    └─tokenAnalysis //词法分析的源代码包
```

1.2 代码运行方式

方法 1:

运行 CompilerTest.java，程序会自动完成整套的词法和语法分析过程。

方法 2:

①首先运行 src/tokenAnalysis 下的 test.java，自动对所给的测试文件进行词法分析，得到 token 输出并存放在 outputFile/tokenAnalysisOutput 文件夹下，并用可唯一标识的文件名进行命名整理。②然后运行 src/syntaxAnalysis 下的 test.java，自动对 outputFile/tokenAnalysisOutput 下的 token 文件进行语法分析，并将结果存

储在 `outputFile/syntaxAnalysisOutput` 文件夹下。如果语法分析出错，分析出错前的语法树会被输出到文件中，并且最后一行输出错误行数及相关信息。如果正确，则输出完整的语法树。

2.自定义测试方案

所有测试用例均为自己根据路径覆盖科学设计，完全覆盖了各产生式对应的抽象语法树中的所有情况，验证了代码实现的完整性和正确性。在代码实现过程中严格依据上下文无关文法编写递归下降程序，各产生式对应的函数编写严谨可靠，经过多次测试尚未发现问题，以下自定义测试方案若涵盖不全，请老师将代码文件放在 `inputFile` 文件夹下，运行程序进行语法分析测试。

2.1 struct_type 测试

对产生式 `struct_type->“struct” ID “{” member_list “}”`进行测试，对应的测试文件为 `inputFile/语法测试文件/自定义测试文件/struct_type*.txt`。

2.1.1 正常代码测试

输入：

```
struct_type.txt - 记事本
文件(F) 编辑(E) 格式(O) 3
struct test
{
    int32 a1;
    int64 a2,a3,a4;
    long double a5;
}
```

输出：

```
syntaxAnalysisResult_自定义测试文件_struct_type.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct_type: struct
  ID: test
  member_list
    member
      type_spec
        base_type_spec
          integer_type
            signed_int: int32
    declarators
      declarator
        ID: a1
    member
      type_spec
        base_type_spec
          integer_type
            signed_int: int64
    declarators
      declarator
        ID: a2
      declarator
        ID: a3
      declarator
        ID: a4
    member
      type_spec
        base_type_spec
          floating_pt_type: long double
    declarators
      declarator
        ID: a5
```

2.1.2 struct 中无 member 测试

输入:

```
struct_type_without_member.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct test
{
}
```

输出:

```
syntaxAnalysisResult_
文件(F) 编辑(E) 格式(O)
struct_type: struct
  ID: test
  member_list
```

按照题目中给的产生式，但没有 member 时，struct_type 的表达式也应该是对的，但是老师给的测试文件中认为这个是错误情况，这是此次实验中的一个问题。

2.1.3 struct 匹配失败测试

输入:

```
struct_type_error1.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V)
strcut test
{
    int8 a1;
}
```

输出:

```
syntaxAnalysisResult_自定义测试文件_struct_type_error1.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct_type: struct
error happens in row 1: ID strcut is not equal to struct
```

程序报错, 指出第一行中的 ID 类型的单词 strcut 与关键字 struct 不匹配

2.1.3 ID 匹配失败测试

输入:

```
struct_type_error2.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V)
struct int8
{
    int16 a1;
}
```

输出:

```
syntaxAnalysisResult_自定义测试文件_struct_type_error2.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct_type: struct
error happens in row 1: RESERVEDWORD int8 doesn't match ID
```

程序报错, 指出第一行中的保留字类型的单词 int8 与 ID 不匹配。

2.2 数据类型测试

对所有的数据类型: char, boolean, unsigned short, uint16, unsigned long, uint32, unsigned long long, uint64, uint8, short, int16, long, int32, long long, int64, int8, float, double, long double 进行测试, 对应的测试文件为 inputFile/语法测试文件/自定义测试文件/data_type.txt。

输入:

```
data_type.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮
struct test
{
    char ch1;
    Boolean b1;
    unsigned short us1;
    unsigned long ul1;
    uint8 ui1;
    uint16 ui2;
    uint32 ui3;
    uint64 ui4;
    unsigned long long ui5;
    int8 i1;
    int16 i2;
    int32 i3;
    int64 i4;
    short s1;
    long l1;
    long long ll1;
    float f1;
    double d1;
    long double ld1;
}
```

输出:

```
syntaxAnalysisResult_自定义测试文件_data_type.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
type_spec
  base_type_spec
    integer_type
      signed_int: long long
  declarators
    declarator
      ID: ll1
member
  type_spec
    base_type_spec
      floating_pt_type: float
  declarators
    declarator
      ID: f1
member
  type_spec
    base_type_spec
      floating_pt_type: double
  declarators
    declarator
      ID: d1
member
  type_spec
    base_type_spec
      floating_pt_type: long double
  declarators
    declarator
      ID: ld1
<
Windows (CRLF) 第 148 行, 第 38 100%
```

2.3 其他表达式综合测试

其他表达式指与 `struct_type` 和基本的数据类型无关的表达式测试，如 `exp_list`, `or_expr`, `xor_expr` 等。

编写代码，覆盖所给文法中所有可能出现的情况（有 `repeat` 的表达式出现至少两次便认为完全覆盖）。

输入:

```
complete_test.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct test{
    float a1;
    long double a5;
    STRUCT warp_int{
        long long i8;
    } int1,int2;
    struct warp_signInt{
        unsigned long s8;
        unsigned long long s9;
    } sign1,SIGN2;
    char a_6["compile"+"studying"," ","abs\b\t\n\f\r\"\\abs"];
    BOOLEAN a_bool_7[10^2,1&2&3,TRUE|FALSE,~FALSE|~TRUE];
    long a8[1024>>10,0<<10,100*2/10%2,100+21-19];
    boolean a9[10^2&3|-1>>10&100<<-10-10+100*2/10%2];
    struct warp_1{
        float w1;
        long w2;
        struct warp_2{
            boolean w3[111&2];
            char w4["\b\t\n\f\r\"\\"];
        } w5,w6;
    } w7,w8;
}
```

输出过长，请见“syntaxAnalysisResult__自定义测试文件_complete_test.txt”

2.4 错误测试

2.4.1 member_list 中“;”不匹配测试

输入：

```
member_list_error.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct test
{
    int16 aaa
    double b;
}
```

输出：

```
syntaxAnalysisResult__自定义测试文件_member_list_error.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct_type: struct
  ID: test
  member_list
    member
      type_spec
        base_type_spec
          integer_type
            signed_int: int16
        declarators
          declarator
            ID: aaa
error happens in row 4: RESERVEDWORD double is not equal to ;
```

程序报错，指出第 4 行中的保留字类型的单词 `double` 与希望匹配的分号“;”不匹配。

2.4.2 base_type_spec 中基本数据类型不匹配测试

输入：

```
base_type_spec_error.txt -
文件(F) 编辑(E) 格式(O) 查
struct test
{
    int128 aaa;
    long bbb;
}
```

输出：

```
syntaxAnalysisResult_自定义测试文件_base_type_spec_error.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct_type: struct
  ID: test
  member_list
  member
    type_spec
    base_type_spec
error happens in row 3: ID int128 doesn't match base_type_spec
```

程序报错，指出第 3 行中的 ID 类型的单词 int128 与基本数据类型 base_type_spec 不匹配。

2.4.3 exp_list 中“]”不匹配测试

输入：

```
exp_list_error.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct test
{
    int16 aaa["compile"+"studying";
    long bbb;
}
```

输出：

```
syntaxAnalysisResult_自定义测试文件_exp_list_error.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct_type: struct
  ID: test
  member_list
  member
    type_spec
    base_type_spec
    integer_type
    signed_int: int16
  declarators
  declarator
    ID: aaa
    exp_list
    or_expr
    xor_expr
    and_expr
    shift_expr
    add_expr
    mult_expr
    unary_expr
    STRING: "compile"
    +mult_expr
    unary_expr
    STRING: "studying"
error happens in row 3: SPECIFICSYMBOL ; is not equal to ]
```

程序报错，指出第 3 行中的专用符号“；”与“]”不匹配。

2.4.4 unary_expr 中(INTEGER|STRING|BOOLEAN)不匹配测试

输入：

```
unary_expr_error.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct test
{
    int16 aaa["compile"+float];
    long bbb;
}
```

输出：

```
syntaxAnalysisResult_自定义测试文件_unary_expr_error.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
struct_type: struct
  ID: test
  member_list
    member
      type_spec
        base_type_spec
          integer_type
            signed_int: int16
          declarators
            declarator
              ID: aaa
              exp_list
                or_expr
                  xor_expr
                    and_expr
                      shift_expr
                        add_expr
                          mult_expr
                            unary_expr
                              STRING: "compile"
                            +mult_expr
                              unary_expr
error happens in row 3: RESERVEDWORD float doesn't match (INTEGER|STRING|BOOLEAN) in unary_expr
```

程序报错，指出第 3 行中的保留字类型的单词 `float` 与 `INTEGER|STRING|BOOLEAN` 均不匹配。