Project 2. Parser

2016025823

JUNG JAE, LEE

1. Compilation method and environment

* Edit - MaxOS, VisualStudio Code
* Run – Virtual Box, Ubuntu 16.04LTS
* Manage – Github(in progress), Gitlab(done)
* HowtoRun
  + yacc -d cminus.y 🡪 y.tab.c 생성
  + make 🡪 Compile and link
  + ./cminus ./test\_file\_name 🡪 Execute

1. Explanation about how to implement and how to operate.

* 텍스트이(가) 표시된 사진

  자동 생성된 설명텍스트이(가) 표시된 사진

  자동 생성된 설명Globals.h – cminus.y에서 사용할 추가적인 node들의 kind를 정의하였다. 또한 tree, node 구조체에 INT,VOID 등을 저장한 type과 array 선언에 사용될 nam과 size를 멤버변수로 가지는 구조체를 추가하였다.

Figure 1. NodeKind, Arr\_info struct 선언

Figure 2 treeNode 구조체

* main.c – 명세와 동일하게 수정.
* Makefile – 명세와 동일하게 작성하였으나, analyze.o cgen.o 등의 파일에서 에러가 발생하였고, 해당 오브젝트 파일을 제외하고 컴파일하도록 수정하였다.
* cminus.y – 주어진 명세의 BNF Grammar for C-minus 와 동일하게 구현하였고, 아래와 같은 문법을 추가하였다.
  + 텍스트이(가) 표시된 사진

    자동 생성된 설명Identifier, num – 기존 존재하던 변수인 savedName 아래, savedNumber 변수를 추가하였다. 그러나 이와 같은 변수들을 추가적인 토큰을 처리할 때, 데이터가 overwrite 되기 때문에, 아래와 같이 구현하고 다른 토큰에서는, identifier와 num을 호출하게 하였다.

Figure 3 Overwrite 방지

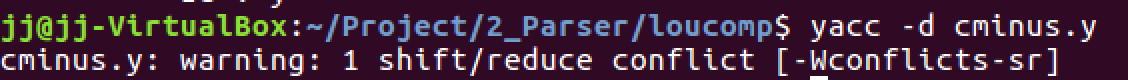
* + If 관련 shift reduce conflict – 기존 명세를 따라 구현하였으나 “yacc -d cminus.y” 수행시 아래와 같은 conflict가 발생하였다. 따라서 “%nonassoc” 와 “%prec” 등을 활용하여 명세와 같은 if간의 우선순위를 정의하였다.

Figure 4 Conflict 발생

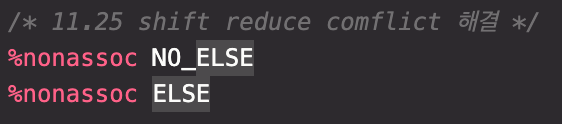


Figure 5 nonassoc 추가

텍스트이(가) 표시된 사진

자동 생성된 설명

Figure 6 우선순위 부여

* + 텍스트이(가) 표시된 사진

    자동 생성된 설명배열 선언 – global.h에서 선언하였던 treeNode 구조체안에 선언된 Arr\_info 구조체에 데이터를 저장하였다.

Figure 7 변수 선언

* + 텍스트이(가) 표시된 사진

    자동 생성된 설명Type – INT, VOID 와 같은 정보를 저장하기 위하여 아래와 같은 문법을 추가하였다.

Figure 8 Type

* 텍스트이(가) 표시된 사진

  자동 생성된 설명util.c – 기존에 newStmtNode()와 newExpNode() 함수만 존재하는데, global.h에서 추가한 Decl, Param 등의 NodeKind 에 대하여 동일한 함수를 작성하였다. 또한, printTree에도 반영하였고, 출력 형식을 명세와 동일하도록 작성하였다..

Figure 9 PrintToken에 추가

Figure 10 new (Kind) Node

텍스트이(가) 표시된 사진

자동 생성된 설명

1. Example and Result Screenshot

* 텍스트이(가) 표시된 사진

  자동 생성된 설명Test1 – Function declaration, parameter, Compound statement 등이 정상적으로 Parsing 되었고, 함수와 변수의 이름과 자료형이 원활하게 출력되었다.

Figure 11 Test1

* 텍스트이(가) 표시된 사진

  자동 생성된 설명Test2 – 배열 선언시, 해당 배열의 size를 잘 저장하고 있음을 알 수 있다.

Figure 12 Test2

* 텍스트, 테이블이(가) 표시된 사진

  자동 생성된 설명Test3.result – 명세에서 주어진 코드를 테스트하였고, else가 가까운 if와 associate되는 것을 알 수 있다.

Figure 13 if 관련 명세

텍스트이(가) 표시된 사진

자동 생성된 설명

Figure 14 Test3