

PL Assignment #8: Cute15 Built-in Function 구현 - 2

과제물 부과일 : 2015-05-04 (월)

Program Upload 마감일 : 2015-05-10 (화) 23:59:59

문제

Cute15 문법에 따라 작성된 program이 as08.txt에 저장되어 있다. 이 프로그램은 Cute15의 리스트를 조작하며 그 결과를 구하고 있다. 이러한 프로그램을 input file로 하여, syntax tree를 출력하고 프로그램 다양한 연산자들을 사용하여 수행 결과를 출력하는 프로그램을 작성하시오.

Cute15의 built-in function

이번 과제에서 구현해야하는 연산자들은 다음과 같다.

- 산술 연산

"+" | "-" | "*" | "/"

예) (+ 1 2)

3

- 관계연산

"<" | "=" | ">"

예) (> 1 5)

#F

- 논리 연산

"not"

예) (not #F)

#T

- 조건문

"cond"

예) (cond ((> 1 2) 0) (#T 1))

추가설명

- 조건문을 위한 cond 는 다음과 같은 구조를 가진다.

```
(COND
  (condition1 result1 )
  (condition2 result2 )
  . . .
  (#T resultN ) )
```

이것을 C와 비슷한 문법으로 표현하면 다음과 같다.

```
if ( condition1 )
  return result1
```

```

else if (condition2)
    return result2
else if (condition3)
    ...
else
    return result

```

수행 예시

입력파일

```

(+ 1 2)
(- (+ 1 2) 4)
(> 1 5)
(cond ((null? `(1 2 3)) 1) (( > 100 10) 2) (#T 3) )

```

출력파일

```

3
-1
#F
2

```

Programming

앞서 과제 7에 연산자 처리를 위한 함수를 추가한다.

1. 노드의 자료구조 : 과제 7과 동일
2. parsing : 과제 7과 동일
3. 연산자 구현 (이번 과제의 핵심)

```

private Node runBinary(BinarayOpNode node){
    Node result;
    Node left = node.getNext();
    Node right = left.getNext();

    left = runExpr(left);
    right = runExpr(right);

    if(left ==null || right ==null) errorLog("runBinary runExpr null");
    if(!(left instanceof IntNode)||!(right instanceof IntNode)){
        errorLog("Type Error!");
        return null;
    }
}

```

```

        switch(node.value){
        case MINUS:
            result = new IntNode();
            ((IntNode) result).value = ((IntNode)left).value-
            ((IntNode)right).value;
            return result;
            // 기타 연산자, MINUS와 비슷하게 각자 구현할 것
            // 관계연산은 TRUE, FALSE반환
        }

        return null;
    }
}

```

4. 기존 함수 수정

```

private Node runList(ListNode list) {
    // list의 value가 QuoteNode일 경우
    if (list.value instanceof QuoteNode)
        return runQuote(list);

    Node opCode = list.value;
    if (opCode == null)
        return list;
    if (opCode instanceof FunctionNode)
        return runFunction((FunctionNode) opCode);
    if(opCode instanceof BinarayOpNode)
        return runBinary((BinarayOpNode)opCode);
    return list;
}

private Node runFunction(FunctionNode func) {
    ...

    switch (func.value) {
    ...
    ...

    case NOT:
        ...
        //구현..
        break;
    case COND:
        //구현..
        break;
    default:
        break;

    }

    return result;
}
}

```

5. 테스트

```

// 과제 7과 동일
public static void main(String[] args) throws Exception {
    read file..
    parsing..
    print...

    CuteInterpreter i = new CuteInterpreter();
    for(Node n = node; n != null; n = n.getNext()){
        Node ir = i.runExpr(n);
        pt.println();
        pt.printNode(ir);
    }
}

```