

PL Assignment #7: Cute17 Built-in Function 구현 - 2

과제물 부과일 : 2017-05-04 (목)

Program Upload 마감일 : 2017-05-10 (수) 23:59:59

문제

Cute17의 문법에 맞게 작성된 프로그램을 interpret하는 프로그램을 구현 하시오.

Cute17의 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-연산

1. run_func() 함수와 마찬가지로 rhs1, rhs2를 갖고 온다.(rhs 는 right hand side의 약자)
2. rhs1과 rhs2의 표현을 인터프리트함
3. Operator에 따른 결과를 노드로 생성하여 리턴함

참고자료

```

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, FALSE 반환
    }
    return null;
}

```

Programming-cond

```

private Node runCond(Node param){
    Node cond=null;
    if(((ListNode)param).value instanceof ListNode)
        cond = runExpr(((ListNode)param).value);
    else
        cond = ((ListNode)param).value;
    if(!(cond instanceof BooleanNode)){
        errorLog("Type Error!");
        return null;
    }
    System.out.println(((BooleanNode)cond).value);
    if(!((BooleanNode)cond).value) return runCond(param.getNext());
    return runExpr(((ListNode)param).value.getNext());
}

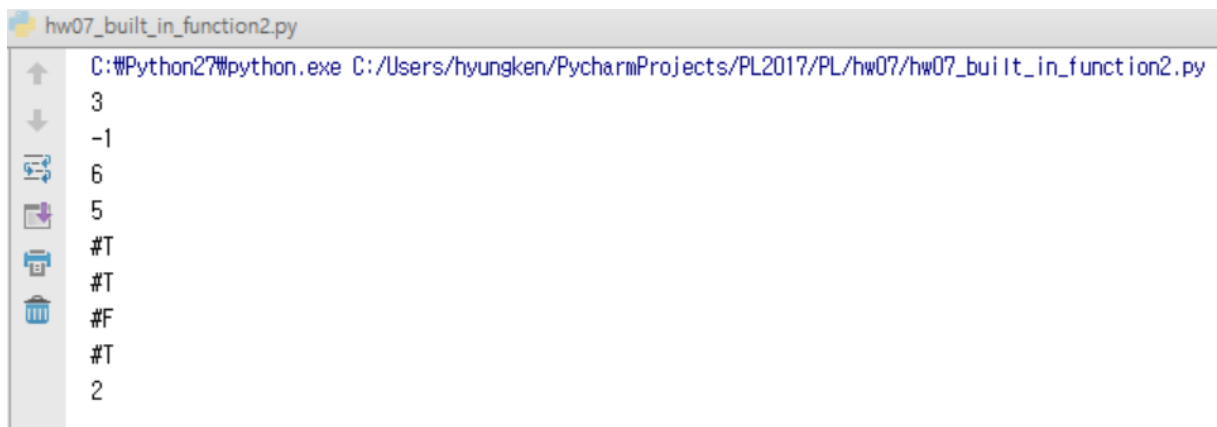
```

테스트

```
def Test_All():
    Test_method("( + 1 2 )")
    Test_method("( - ( + 1 2 ) 4 )")
    Test_method("( * 3 2 )")
    Test_method("( / 10 2 )")
    Test_method("( < 1 5 )")
    Test_method("( = 3 ( + 1 2 ) )")
    Test_method("( > 1 5 )")
    Test_method("( not #F )")
    Test_method("( cond ( ( null? ' ( 1 2 3 ) ) 1 ) ( ( > 100 10 ) 2 ) ( #T
3 ) )")

Test_All()
```

결과화면



The screenshot shows a Python interpreter window titled "hw07_built_in_function2.py". The command prompt shows the execution of the script: "C:\Python27\python.exe C:/Users/hyungken/PycharmProjects/PL2017/PL/hw07/hw07_built_in_function2.py". The output of the script is displayed in the console, showing the results of the Test_All function: 3, -1, 6, 5, #T, #T, #F, #T, and 2.

```
hw07_built_in_function2.py
C:\Python27\python.exe C:/Users/hyungken/PycharmProjects/PL2017/PL/hw07/hw07_built_in_function2.py
3
-1
6
5
#T
#T
#F
#T
2
```