

## 컴파일러 (2015) Assignment 1

제출 기한: 9월8일(화) 밤 23:59:59

사용 언어 : Java

제출 방식: 보고서와 소스코드를 압축하여 업로드.

[분반]학번\_이름.zip ( [00]201501234\_홍길동.zip )

- 보고서 : 분반, 이름, 학번 명시.  
사용된 O/S와 컴파일러 명시.  
문제 해결 방법을 작성하기.  
(소스코드를 따로 첨부하기 때문에,  
보고서에는 소스코드를 과도하게 사용하지 마시고, 필요한 부분만  
프로그램의 동작원리 위주로 작성하세요.)  
개선사항-선택 (불편한 점이나, 앞으로 어떻게 했으면 좋겠다는 의견)
- 소스코드 : 설명이 필요한 부분이나 동작원리를 주석으로 남기기(선택-추천)

다음과 같은 명령으로만 구성된 언어 (가칭 CW) 가 있다.

프로그램은 크게 정의 부분과 출력 부분으로 나뉜다.

반드시 정의 부분이 먼저 다 나오고 끝나면 출력 부분이 나온다.

모든 명령은 각 라인의 첫 칸부터 시작된다.

정의 부분에 올 수 있는 명령은 def 뿐이다.

```
def v := "아무_내용"
    변수 이름 v는 아무_내용을 내용으로 하는 문자열을 값으로 가진다.
    예) def name := "Kim"
    같은 v 가 반복 정의되면 앞에 나왔던 정의는 무시된다.
    아무_내용에는 영문자와 ?, !, 쉼표, 마침표와 빈칸이 들어 갈 수 있다.
    newline은 안 된다.
```

출력 부분의 명령은 아래와 같다. 각 출력 명령의 결과는 명령어마다 새로운 줄에 출력된다.

```
show "아무_내용"
    아무_내용 과 newline을 화면에 출력한다.
    예) show "New Semester"
        ← New Semester를 출력함

show v
    변수 v가 나타내는 문자열과 newline을 출력한다.
    예) show name
        ← 위에 정의된 "Kim"의 내용을 출력함

show Item1, Item2, ... ItemN
    N 개의 문자열과 newline을 출력한다.
    각 Item은 변수일 수도 있고, 문자열일 수도 있다.
    예) show "Hello, ", name, "!"
        ← Hello, Kim! 을 출력함

show N Item
    N 은 숫자이다. Item은 변수일수도 있고 문자열일 수도 있다.
    Item이 나타내게 되는 문자열을 N 번 반복하여 출력하고 newline을 출력한다.
    N 이 음수이면 오류를 낸다.

    예) show 2 name
```

← 앞서 정의된 "Kim"을 연달아 출력한다.

`showD N Delimiter Item`

`show N Item` 과 기본적으로는 동일하나

반복하는 동안 각 출력 사이 사이에 *Delimiter*에 해당하는 문자열을 출력

예) `showD 3 ", " "Hello"`

← Hello,Hello,Hello 를 출력함

`showD Delimiter Item1, Item2, ... ItemN`

`show Item1, Item2, ... ItemN` 과 기본적으로는 동일하나

각 출력 사이 사이에 *Delimiter*에 해당하는 문자열을 출력

예) `show " " "Lee", name`

← Lee Kim 을 출력함

**과제:** 위와 같은 명령을 가지는 CW 소스 파일을 C 코드로 바꾸어 .c 파일에 출력하시오.

예1) 입력 : 파일명 **test.cw**

```
def      n  := "abc"
def      m  := "def"
show     n
show     n, m
show     2  m

...
```

→ 출력: 파일명 **test.c**

```
// 이진 어느 output 이나 미리 존재
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
void main(){
    int i;
    // 여기부터 번역된 프로그램의 시작
```

```
char n[] = "abc";
char m[] = "def";
printf("%s\n", n);
printf("%s%s\n",n,m);
```

```
for (i = 0; i < 2; i++)
    printf("%s", m);
printf("\n");
```

```
...
```

```
.
```

**주의:**

- 제시된 것 이 외의 프로그램 입력의 오류는 없다고 가정한다. (즉 모든 입력이 올바르게 주어 진다고 가정한다.)
- 오류 처리나 오류 메시지를 멋지게 만들 필요는 없다.
- 입력 CW 프로그램은 대소문자 구분이 있다.
- 입력 CW 프로그램에서 빈 줄은 없다고 가정한다.
- 결과로 만든 C 코드의 최적화는 고려하지 않는다.