요약 해석 기법을 이용한 2-Staged 언어의 테스트 커버리지 측정

이 춘 우

SE Lab. 서울대학교

Test Coverage

$$\frac{\textbf{size of executed program with test suite}}{\textbf{size of program to be tested}}$$

```
unction foo(a,b) {
b>0) return 1;
se return 0;
```

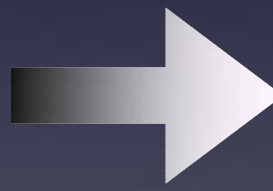
test suite (a,b) {(1,1)}:

decision coverage-**50%**

Test Coverage on Two Staged Language

```
ction foo(a,b) {
x = “if(“+a+”>0) return 3;” + “else
rn 4;”;
>0) return eval(x);
e return 0;
```

(a, b)={(0,0), (0,1

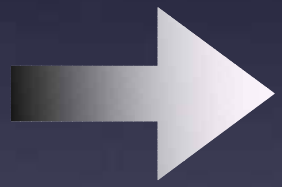


100%

Test Coverage on Two Staged Language

```
ction foo(a,b) {
>0 && a>0) return 3;
e if (b>0) return 4;
e return 0;
```

(a, b)={(0,0), (0,1)}



2/3 = **67%**

새롭게 제안하는
테스트 커버리지 매트릭

Test Coverage on Two-Staged Language

$$\frac{\text{number of branches tested with test suite } (\#B_T)}{\text{number of branches generated in run time } (\#B_e) + \text{number of other branches } (\#B_c)}$$

Example

, j).
= if i then ‘^{1} (f_1 λv.v) else ‘^{2} (f_2 λv.v) in
= if i then ‘^{3} (f_1 λv.v) else ‘^{4} (f_2 λv.v) in
= if i then ‘^{5} (f_1 λv.v) else ‘^{6} (f_2 λv.v) in
= if i then ‘^{7} (f_1 λv.v) else ‘^{8} (f_2 λv.v) in
en
n ‘^{9} (if i then ,$_a$ x else ,$_b$ y)

n ‘^{10} (if i then z else w)

$\#B_c = 2 \times 5 =$
$\#B_e = 2 \times 8 =$

s -> 9[a,b] | 10[
a-> 1|2, b-> 3
c-> 5|6, d-> 7

Example

st Suite	Test Value	DC	NDC
A	{(0,0), (0,1)}	0.6	0.23
B	{(0,0), (0,1), (1,0)}	1.0	0.75
C	{(0,0), (0,1), (1,0), (1,1)}	1.0	0.88

Future Work

- Decision Coverage ==> MC/DC ...
- Test Case Generation for 2-Staged Language