Program Analysis

Data Flow Analysis (Part 6)

Prof. Dr. Michael Pradel

Software Lab, University of Stuttgart

Winter 2020/2021

Outline

- First example: Available expressions
- Basic principles
- More examples
- Solving data flow problems
- Inter-procedural analysis
- Sensitivities 

Sensitivities

Every static analysis: Sensitivities

- Flow-sensitive: Takes into account the order of statements
- Path-sensitive: Takes into account the predicates at conditional branches
- Context-sensitive (inter-procedural analysis only): Takes into account the specific call site that leads into another function

Flow sensitivity : Example

```
if (...) {
        x = 3
        x = 5
}
```

Value of x?

Flow-sensitive : 5

Flow-insensitive : 3 or 5

Path sensitivity: Example

```
x = 0
if (a > 0) {
    x = 1
} else {
    x = 2
}
if (a > 0) {
    x += 3
}
```

← Can x be 5?

Path-sensitive: No

Path-insensitive: Yes

Context sensitivity: Example

```
n = 1
function f(x) {
    if (x) {
        g(3)
    } else {
        n = 3
        g(5)
    }
}
function g(y) {
    ← Can n be equal to y?
}
```

Context-insensitive: Yes
(conflates all call sites of g)

Context-sensitive: No

Quiz: Sensitivities

Consider an intra-procedural data flow analysis (specifically: live variables analysis).

What sensitivities does it have?

Quiz: Sensitivities

Consider an intra-procedural data flow analysis (specifically: live variables analysis).

What sensitivities does it have?

- Flow-sensitive: Yes (every data flow analysis)
- Path-sensitive: No (doesn't track predicates)
- Context-sensitive: Irrelevant (because intra-procedural)

Outline

- First example: Available expressions
- Basic principles
- More examples
- Solving data flow problems
- Inter-procedural analysis
- Sensitivities ✔