Enhancing Control Flow Graph Based Binary Function Identification

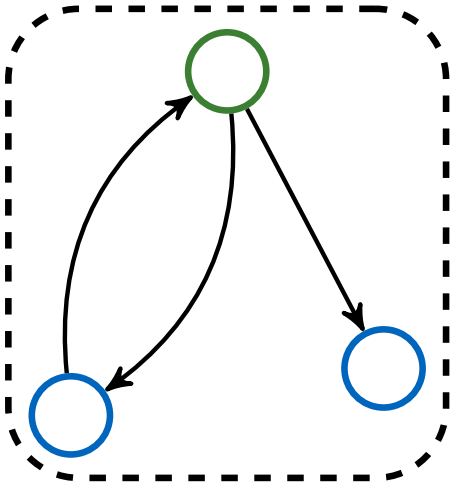
Clemens Jonischkeit
Chair for IT Security

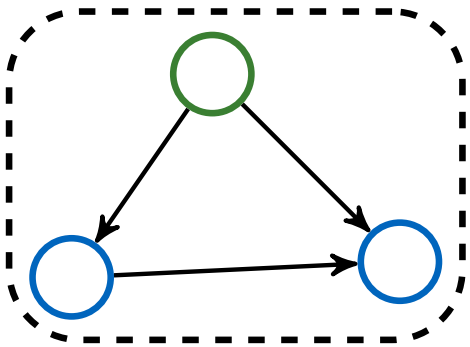
23. November 2017

Motivation

"I just wasted 3 hours of my life...
...because I reversed $foo once again"

Motivation





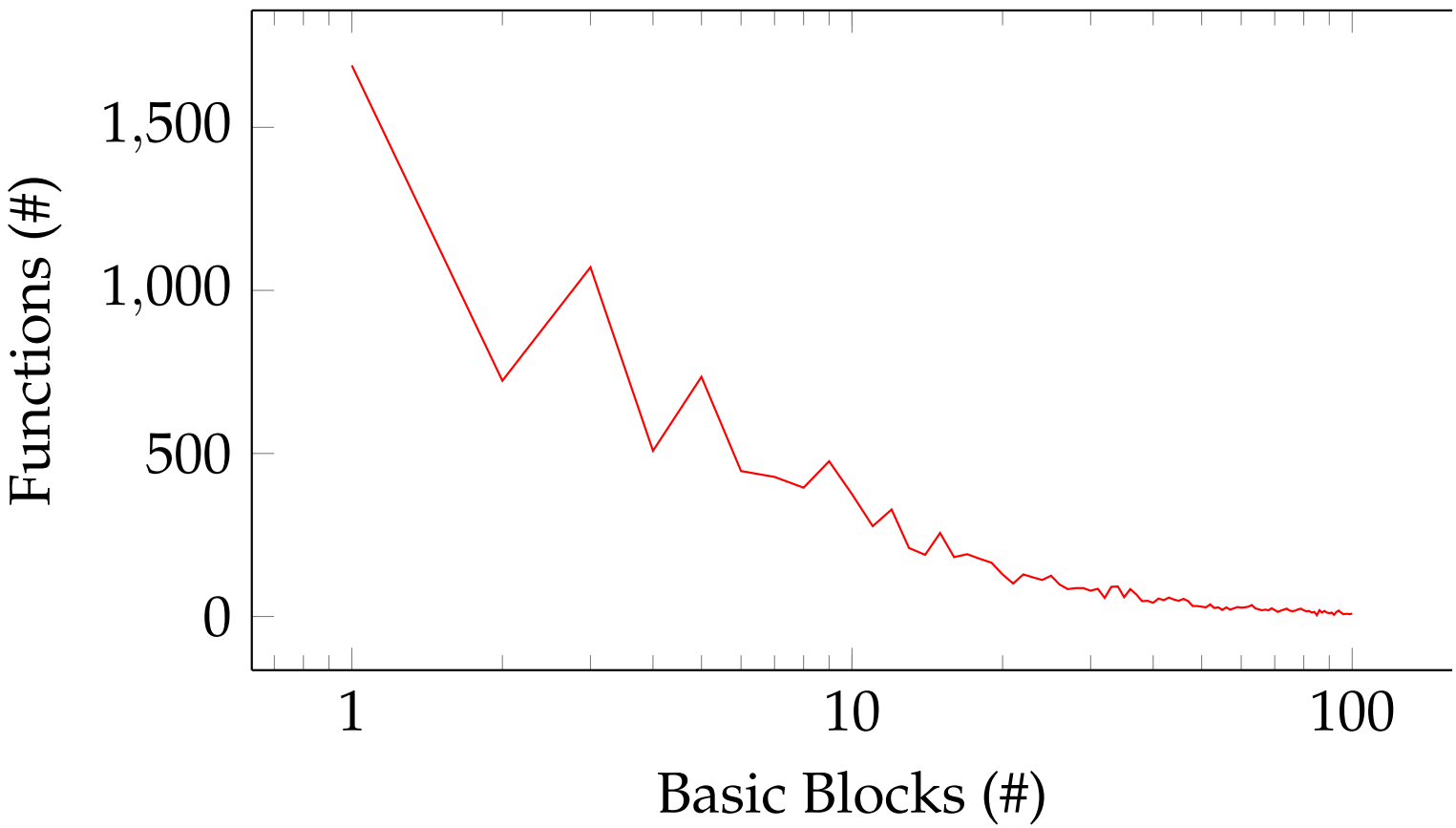
Problem:

- Recover function labels

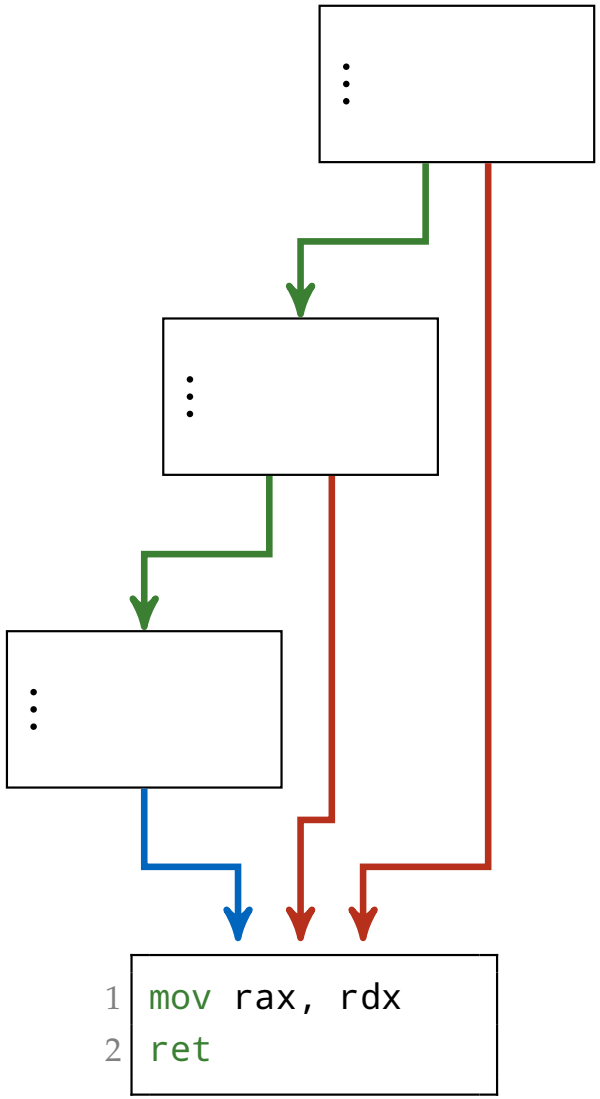
Existing Technology:

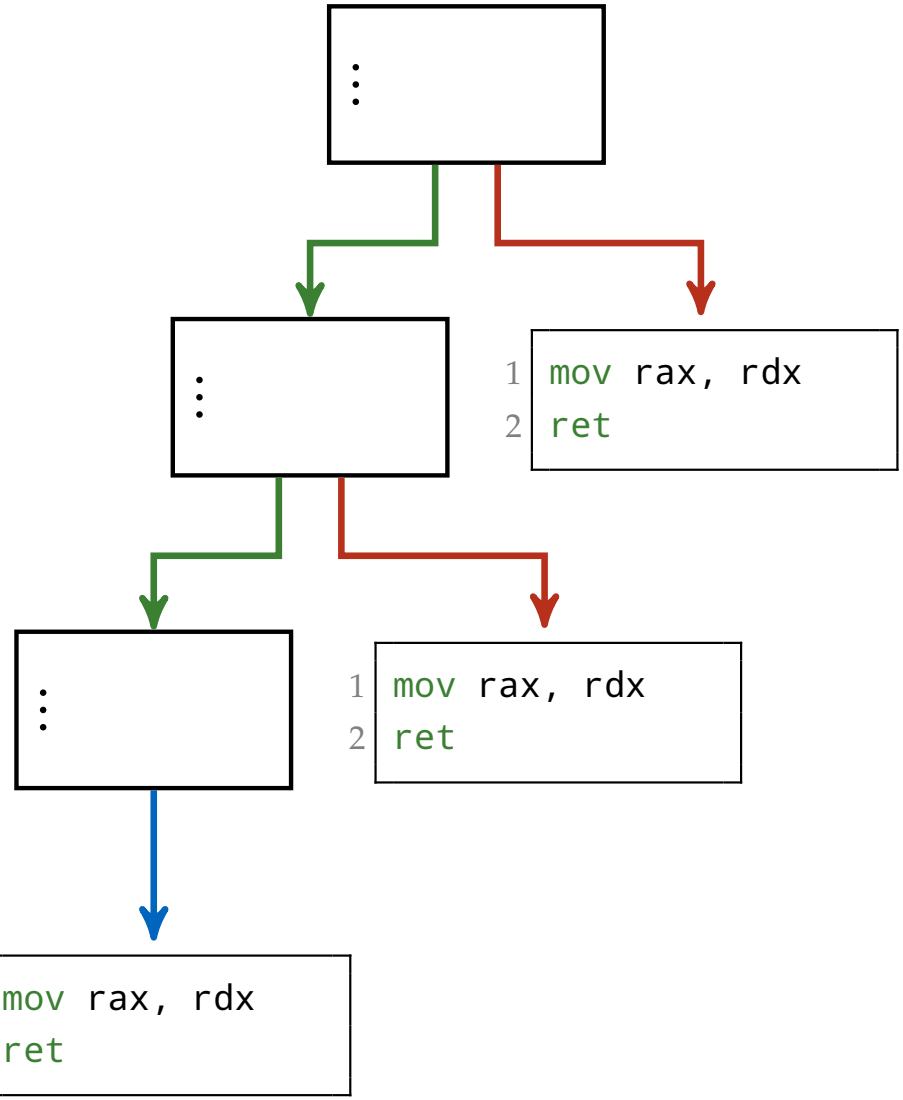
- Pattern Matching (F.L.I.R.T)
- BinDiff
- Diaphora

Problems of CFGs

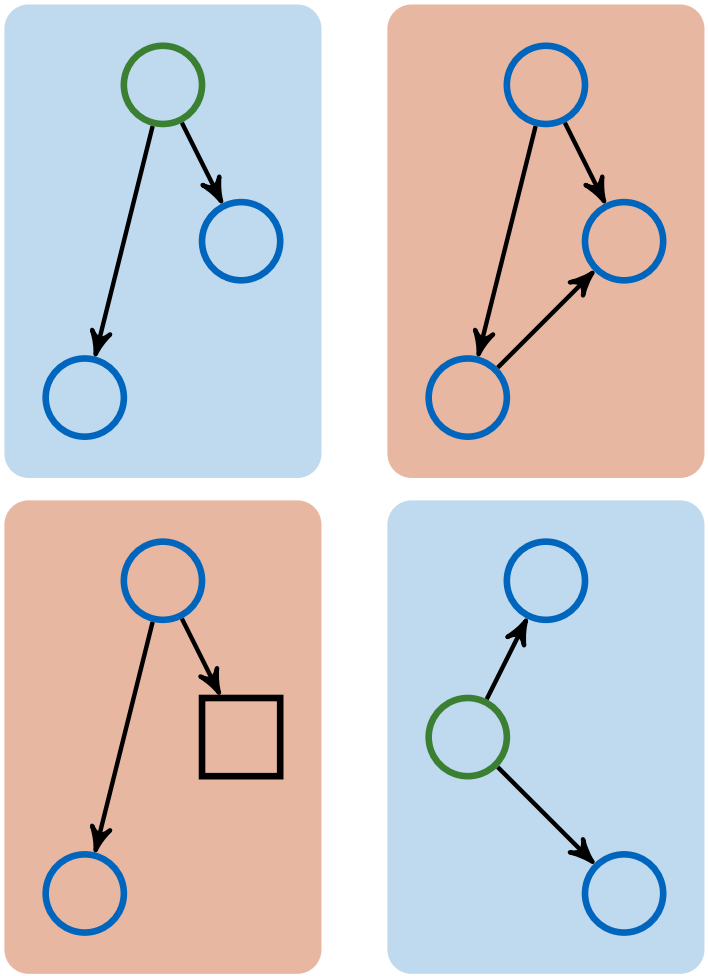


Problems of CFGs





Problems of CFGs



Goal:

- Differentiate similar CFGs
- Resistence against changes

Problems of CFGs

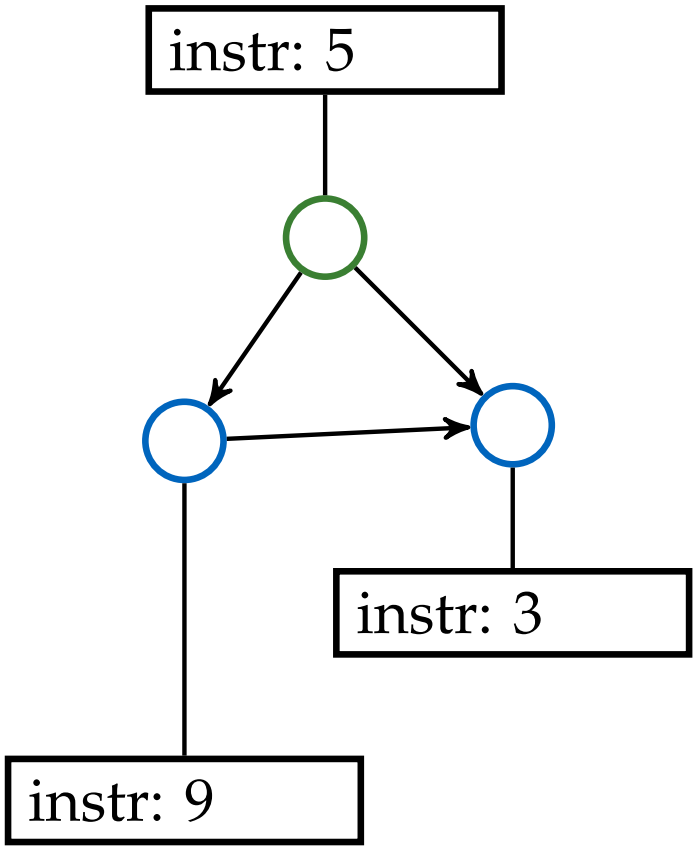
Problem:

- Small Functions: Many functions share the same CFG
- Large Functions: Many different CFGs possible per function

Idea:

- Checking basic block level information
- Normalize CFGs

Normalization



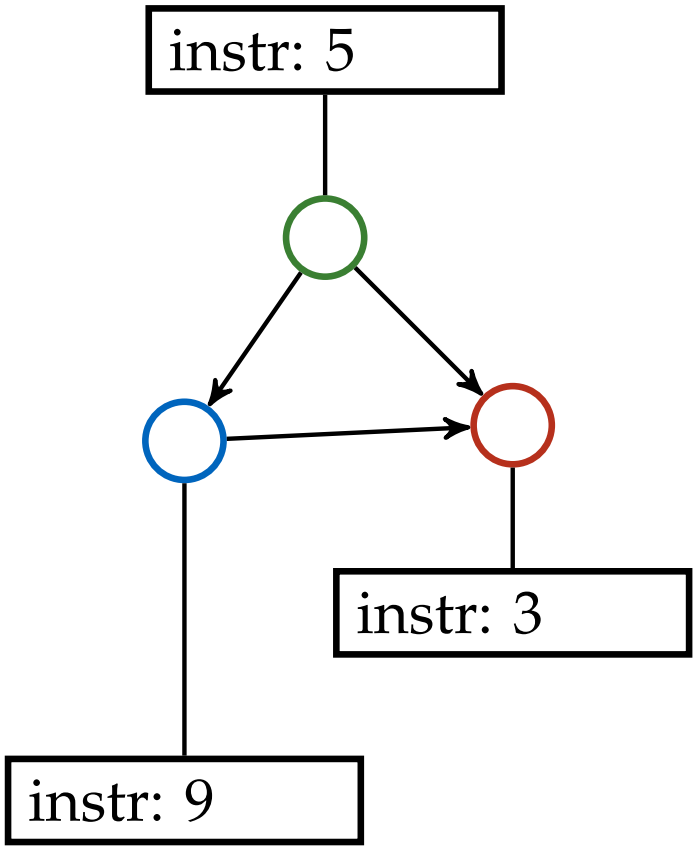
Leaf Inlining:

- Detect Leafs
- Duplicate per parent

Combining Nodes:

- Detect Nodes with one parent
- Filter parents to only have one child
- Combine attributes

Normalization



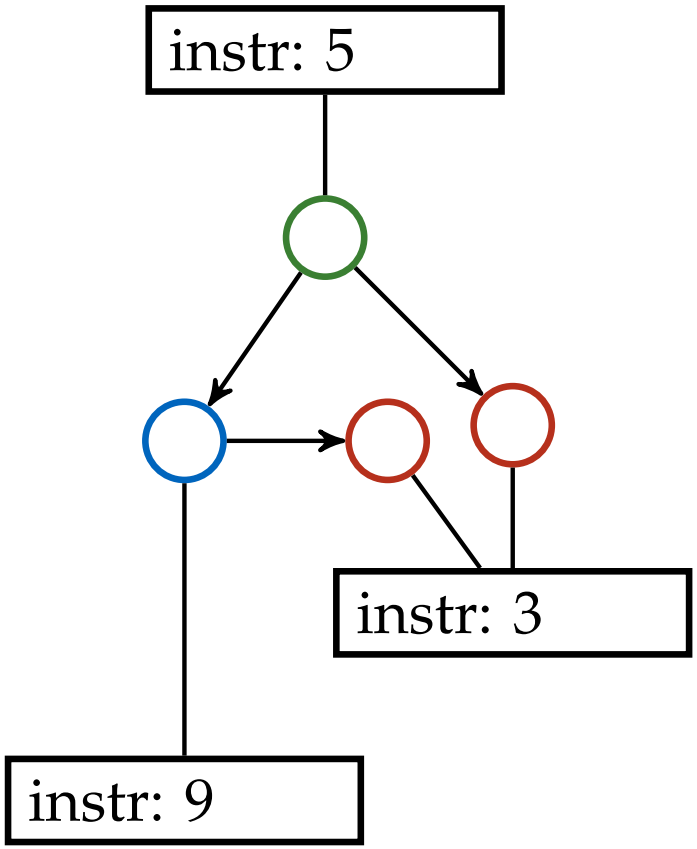
Leaf Inlining:

- Detect Leafs
- Duplicate per parent

Combining Nodes:

- Detect Nodes with one parent
- Filter parents to only have one child
- Combine attributes

Normalization



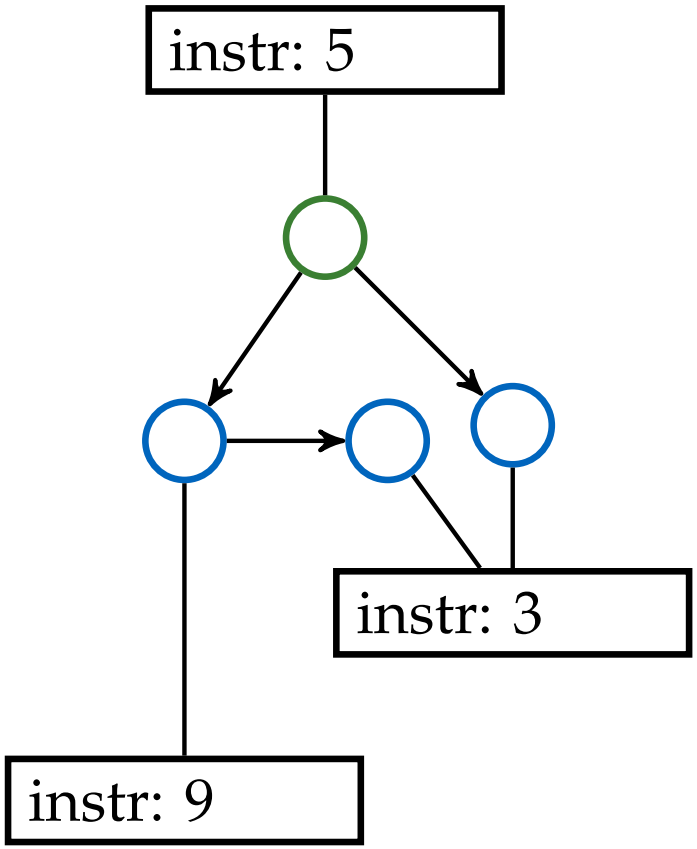
Leaf Inlining:

- Detect Leafs
- Duplicate per parent

Combining Nodes:

- Detect Nodes with one parent
- Filter parents to only have one child
- Combine attributes

Normalization



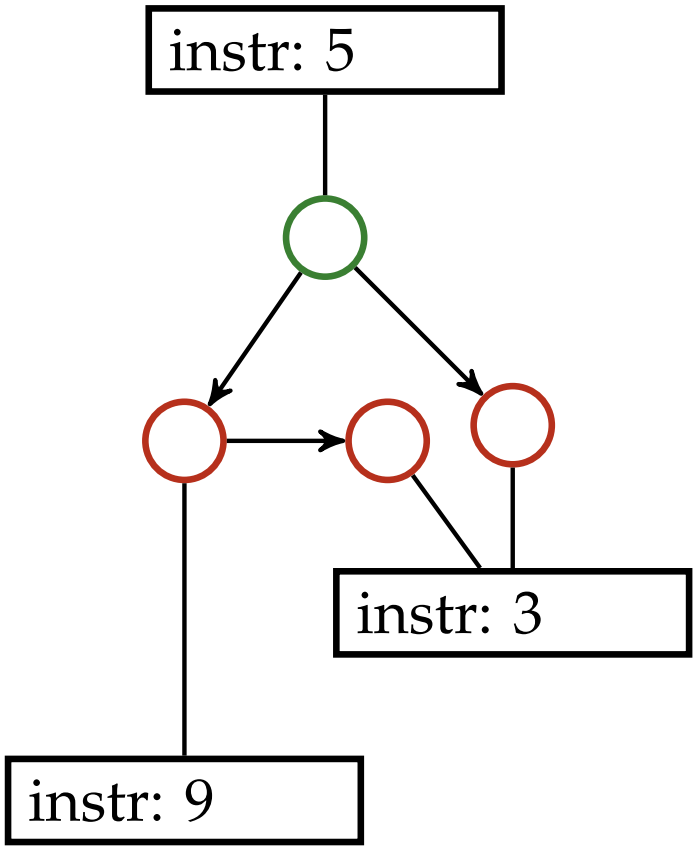
Leaf Inlining:

- Detect Leafs
- Duplicate per parent

Combining Nodes:

- Detect Nodes with one parent
- Filter parents to only have one child
- Combine attributes

Normalization



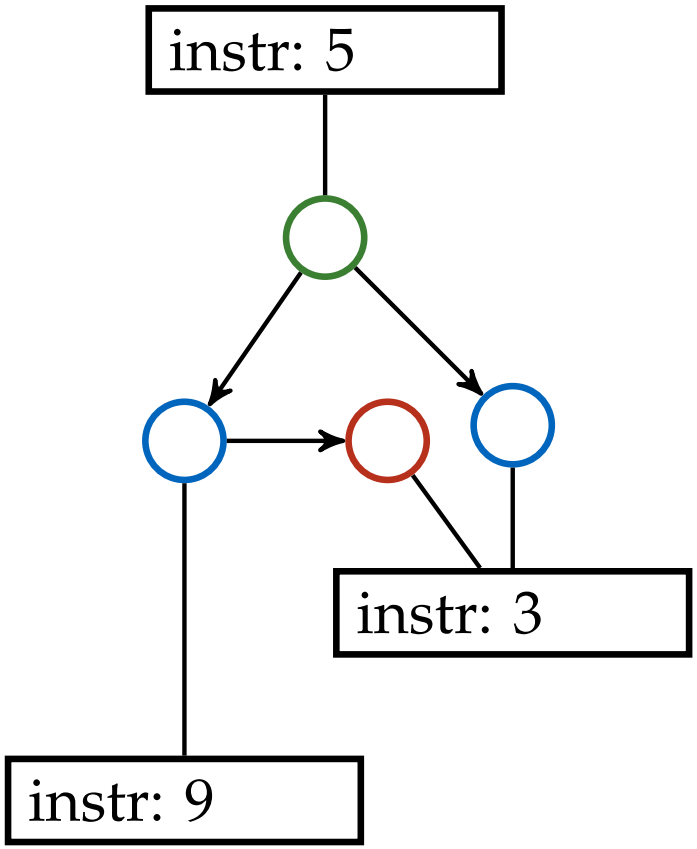
Leaf Inlining:

- Detect Leafs
- Duplicate per parent

Combining Nodes:

- Detect Nodes with one parent
- Filter parents to only have one child
- Combine attributes

Normalization



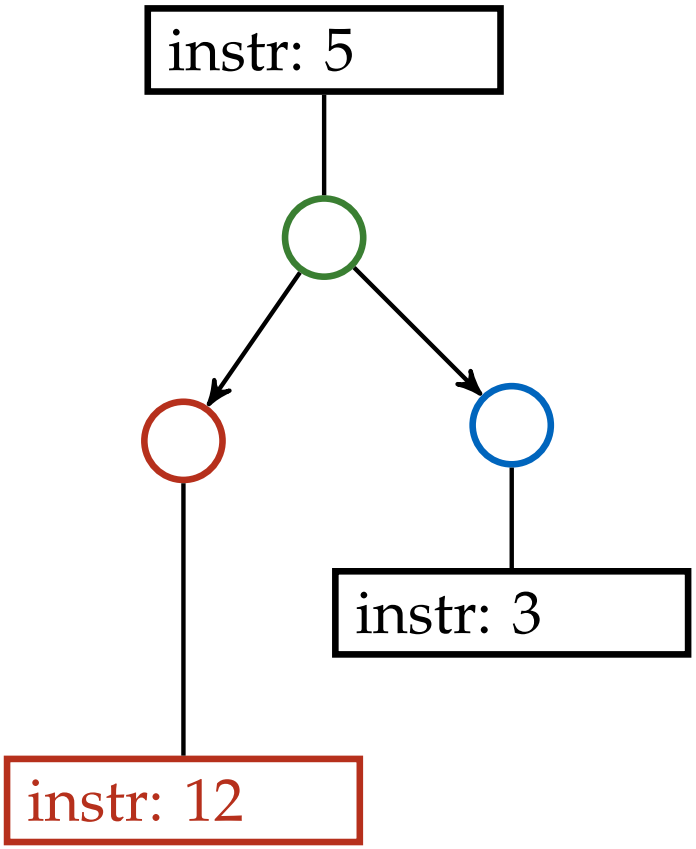
Leaf Inlining:

- Detect Leafs
- Duplicate per parent

Combining Nodes:

- Detect Nodes with one parent
- Filter parents to only have one child
- Combine attributes

Normalization



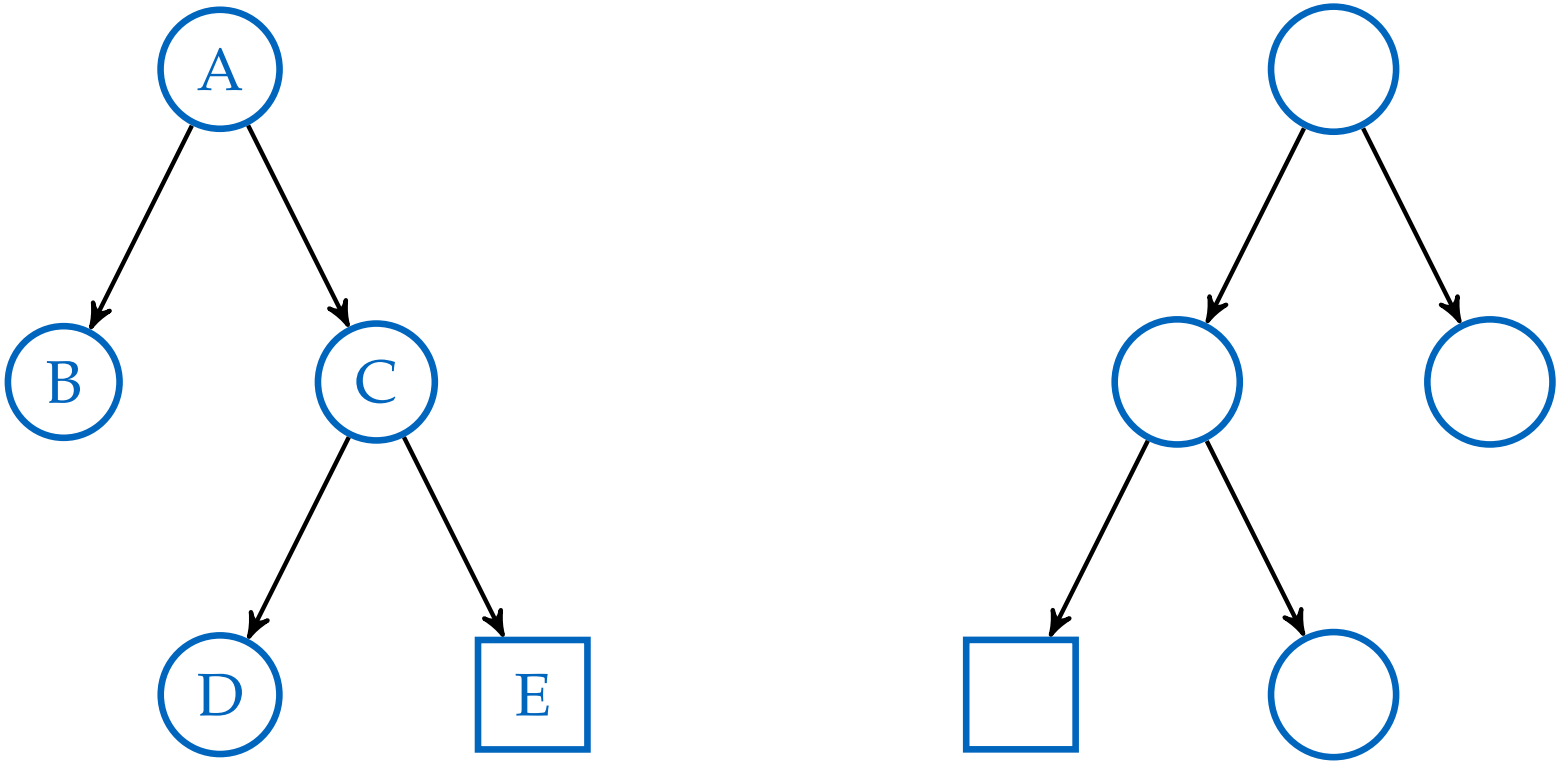
Leaf Inlining:

- Detect Leafs
- Duplicate per parent

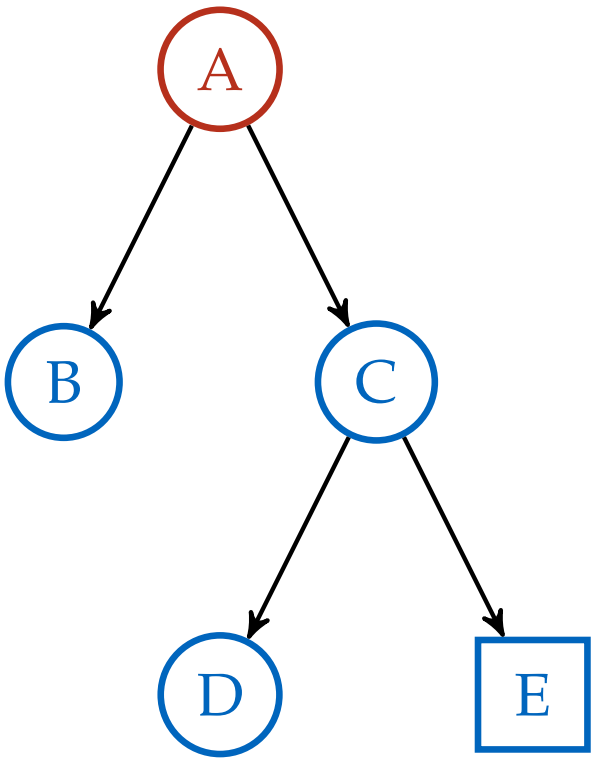
Combining Nodes:

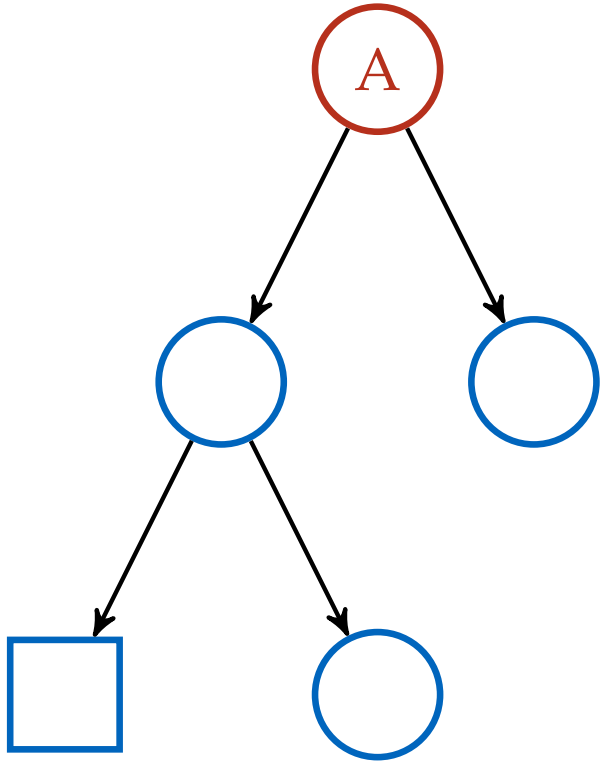
- Detect Nodes with one parent
- Filter parents to only have one child
- Combine attributes

Graph Comparison

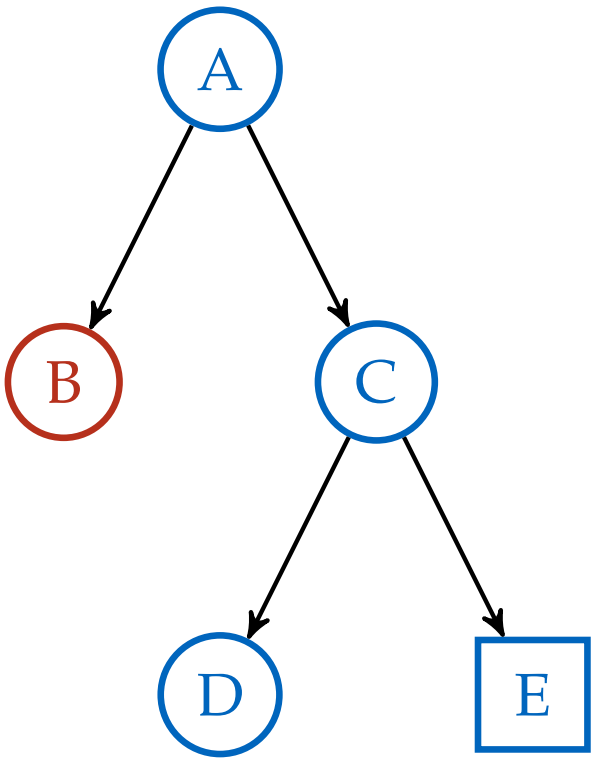


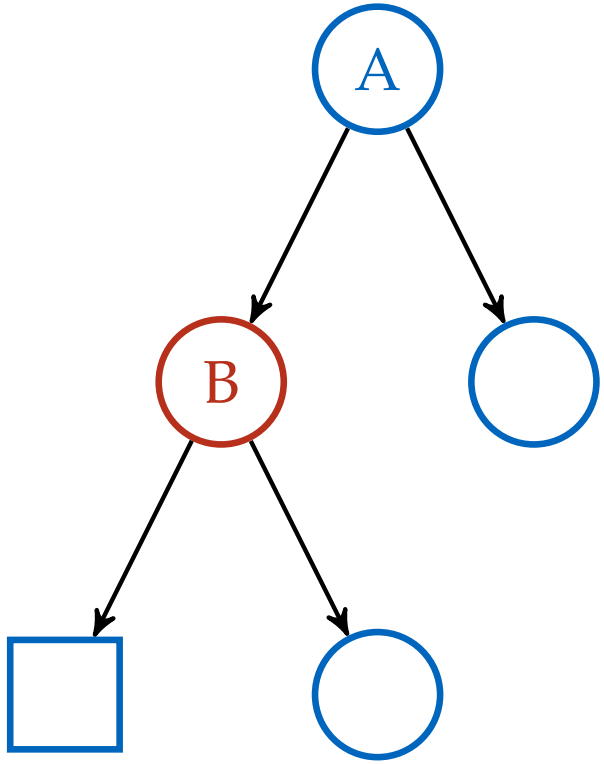
Graph Comparison



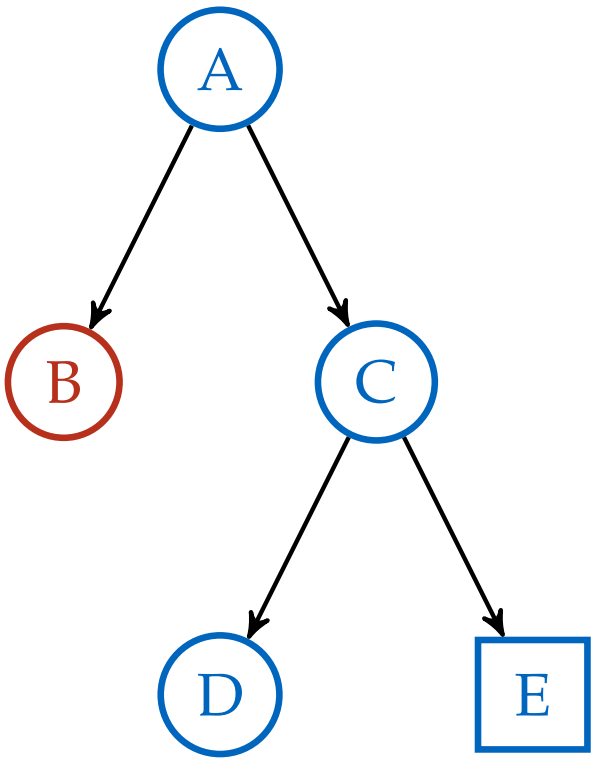


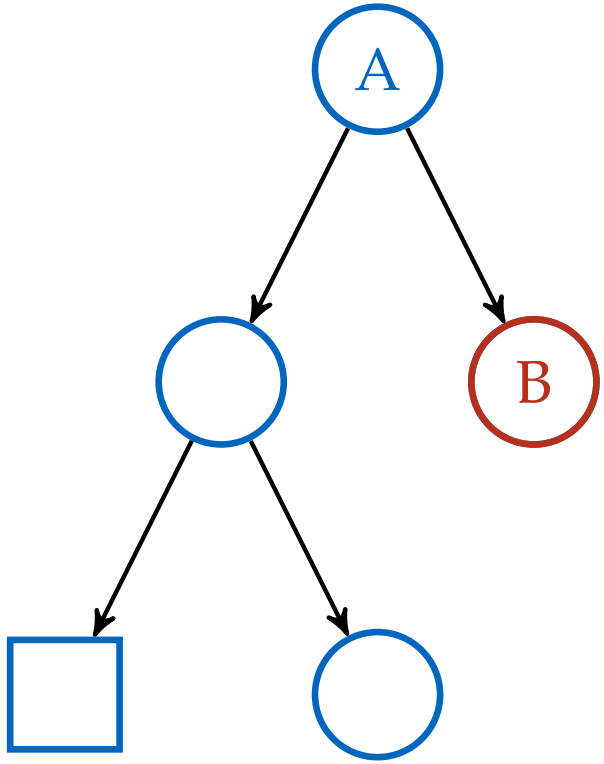
Graph Comparison



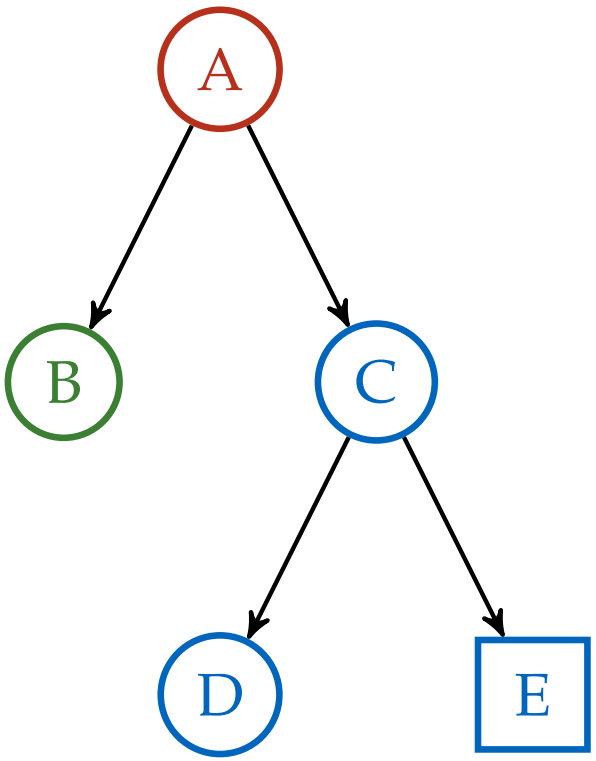


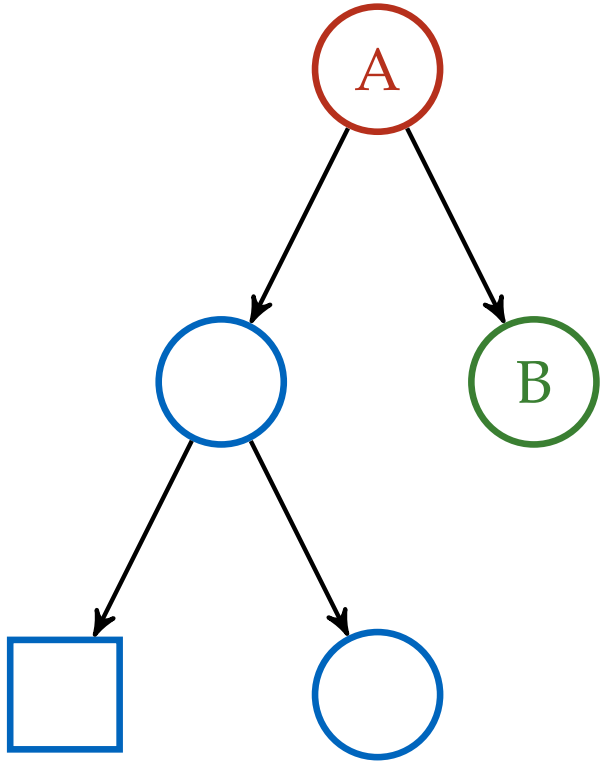
Graph Comparison



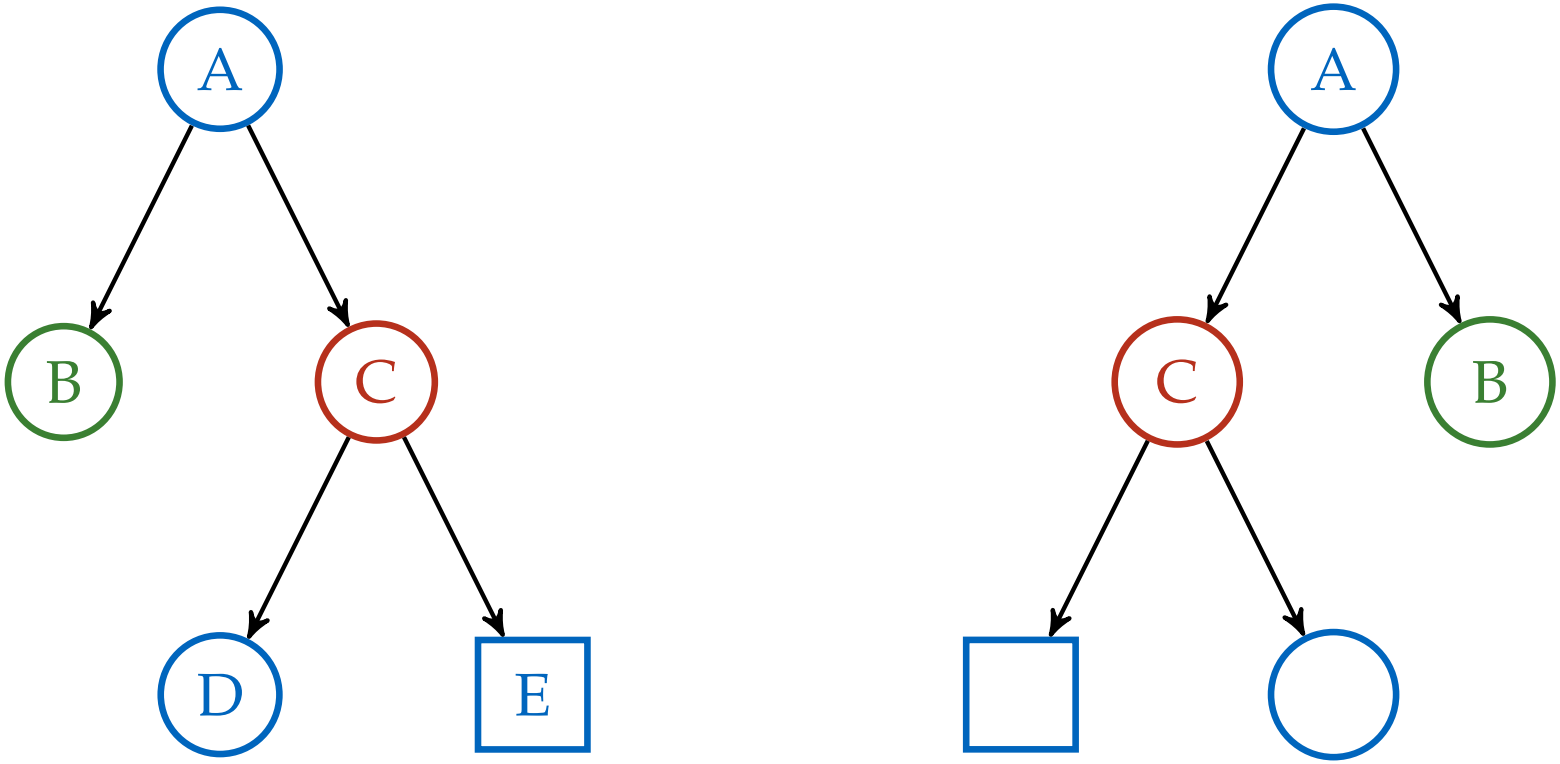


Graph Comparison

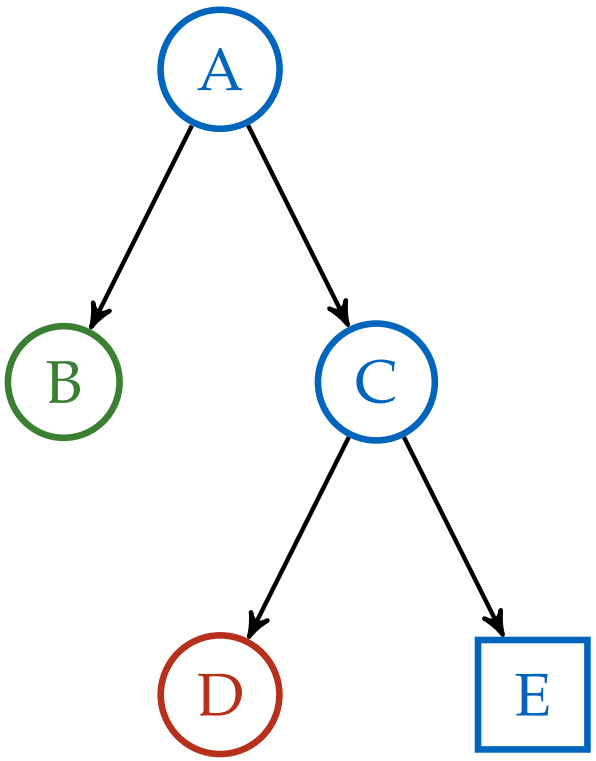


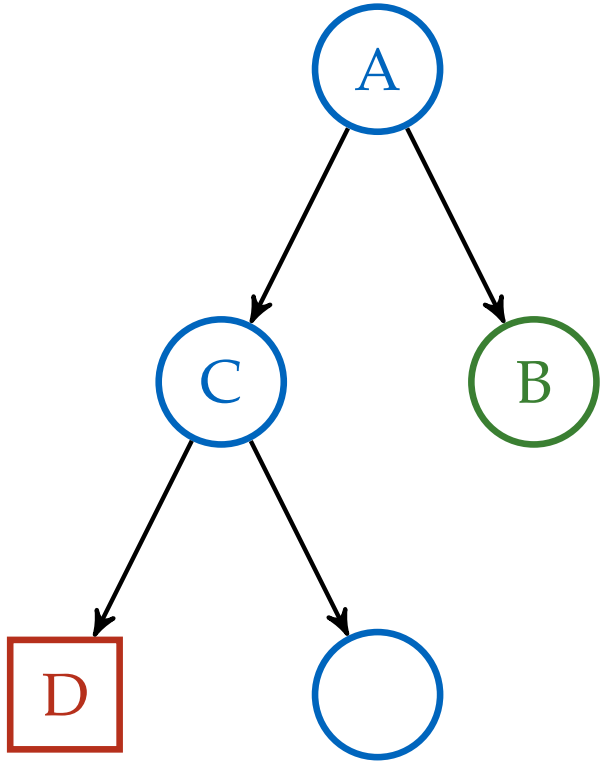


Graph Comparison

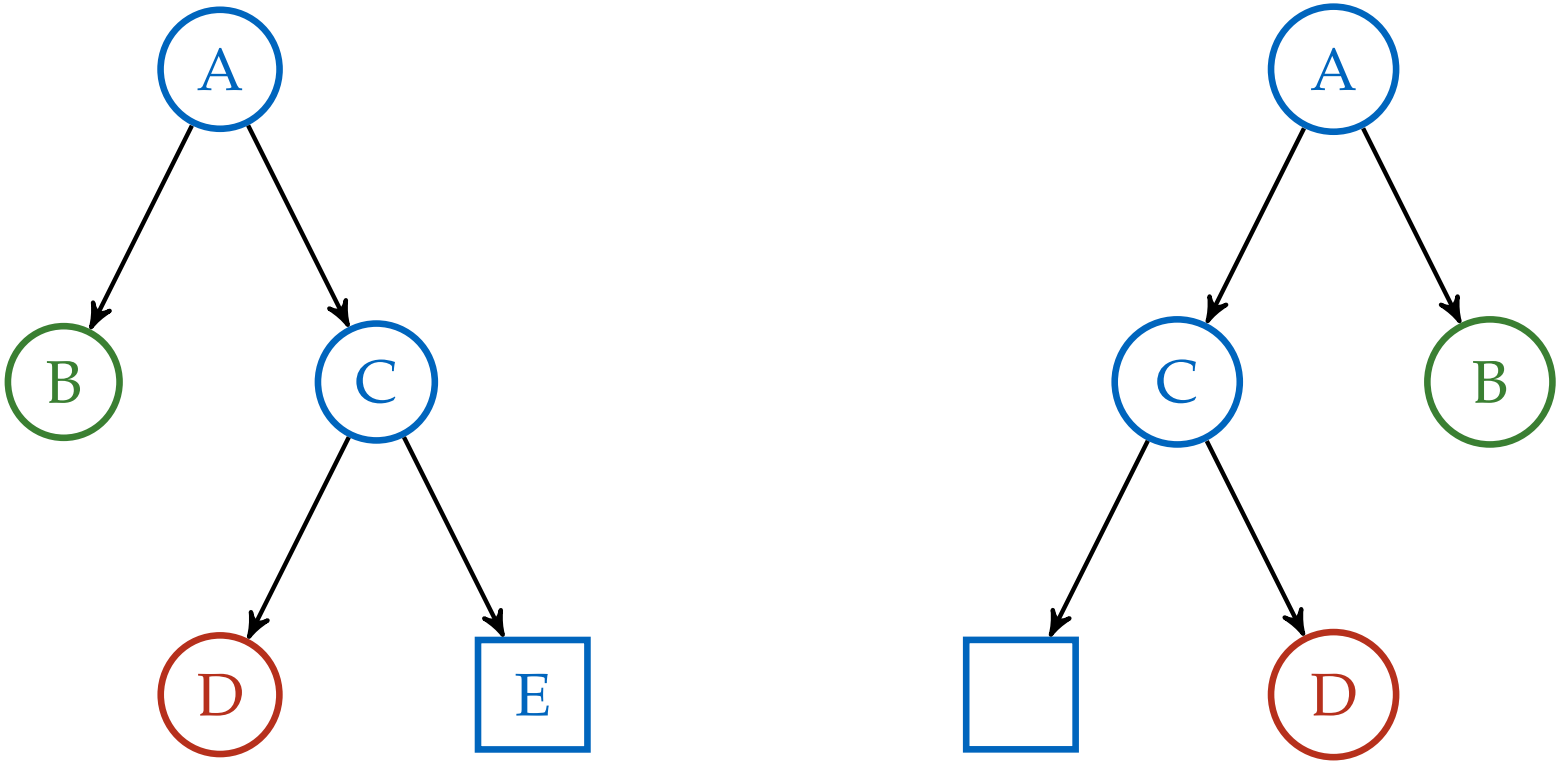


Graph Comparison

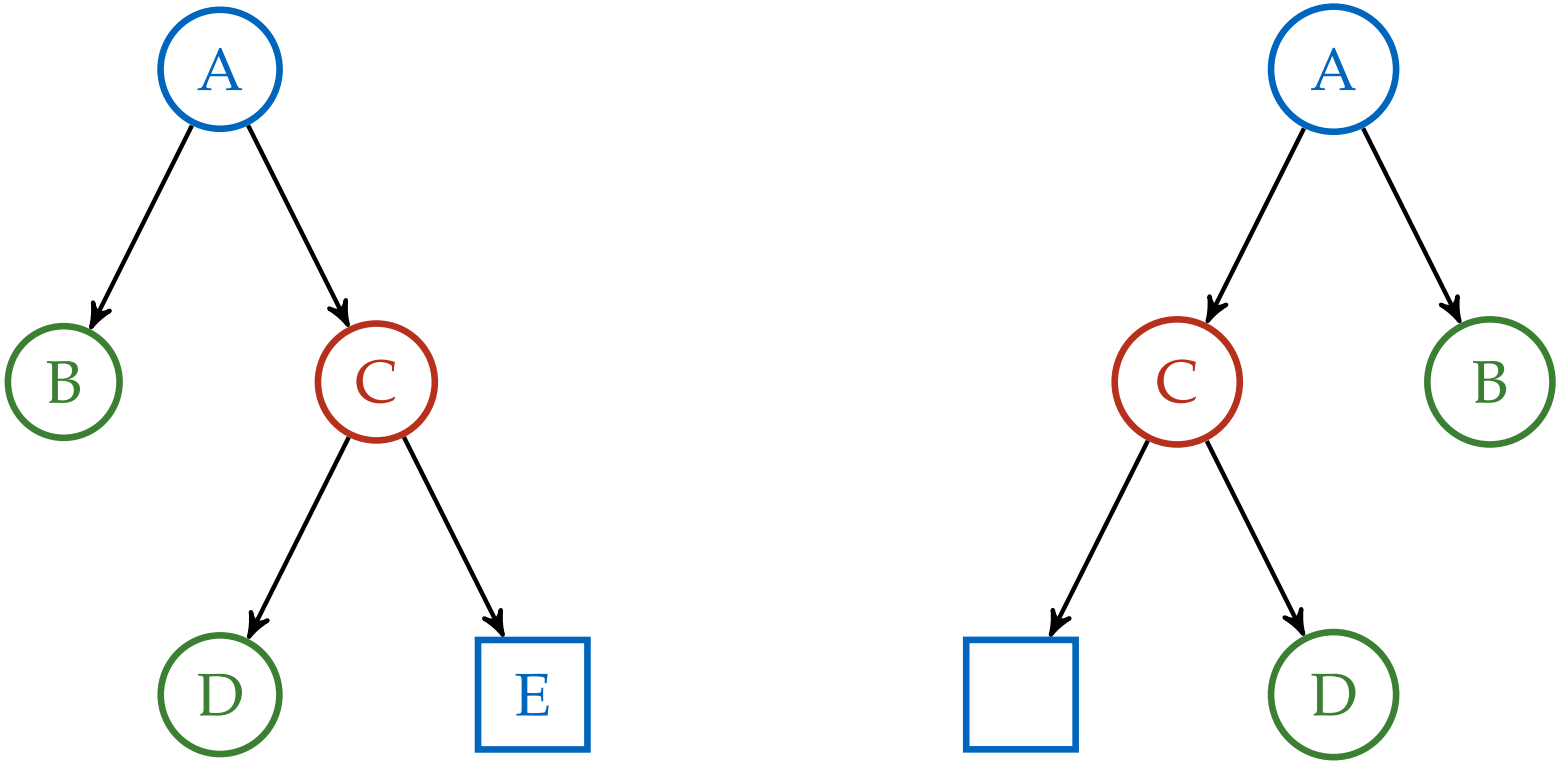




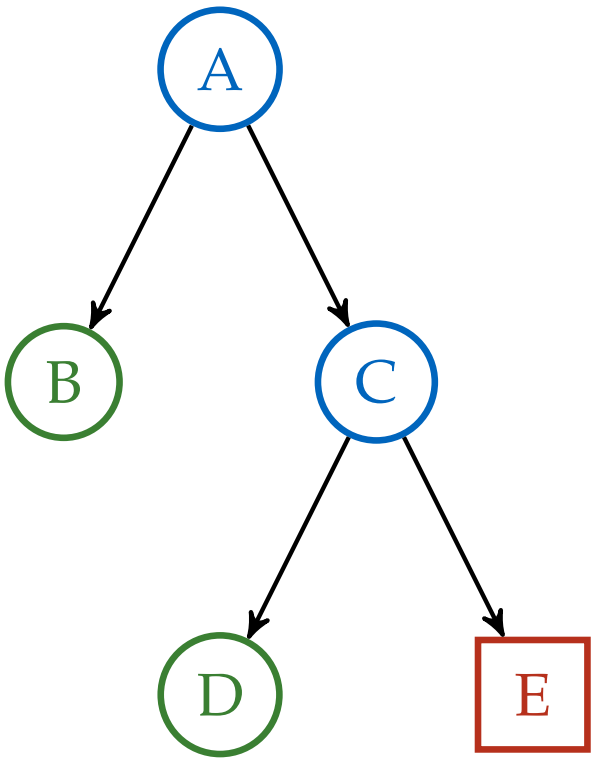
Graph Comparison

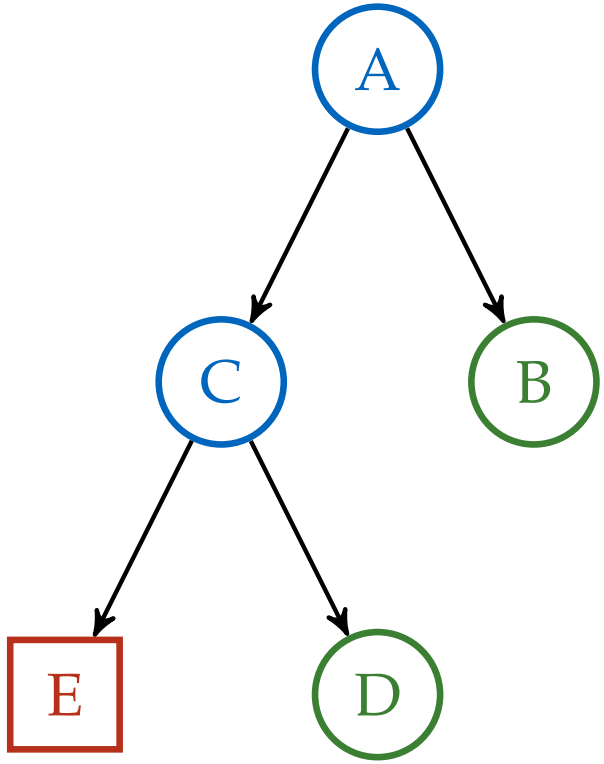


Graph Comparison

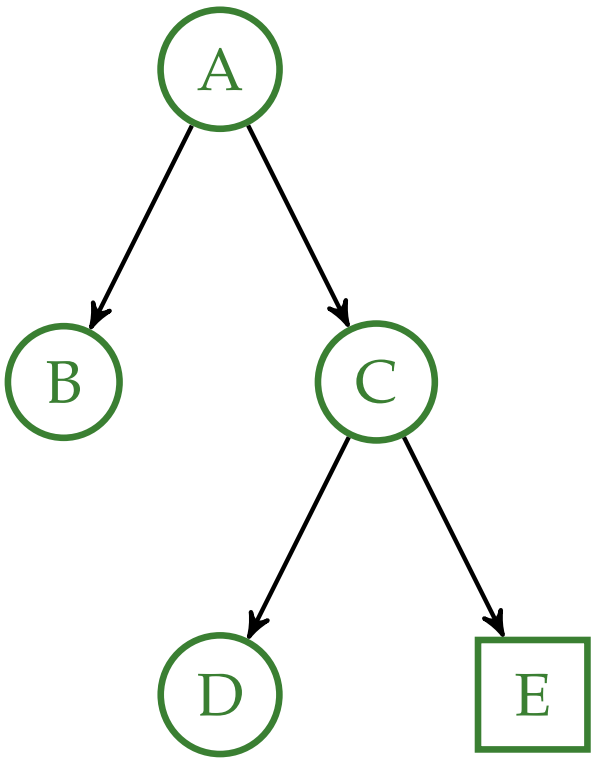


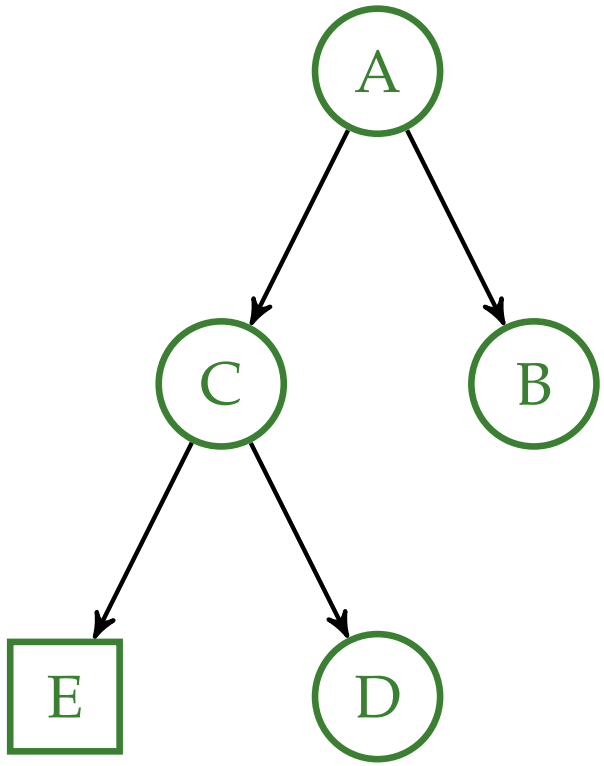
Graph Comparison



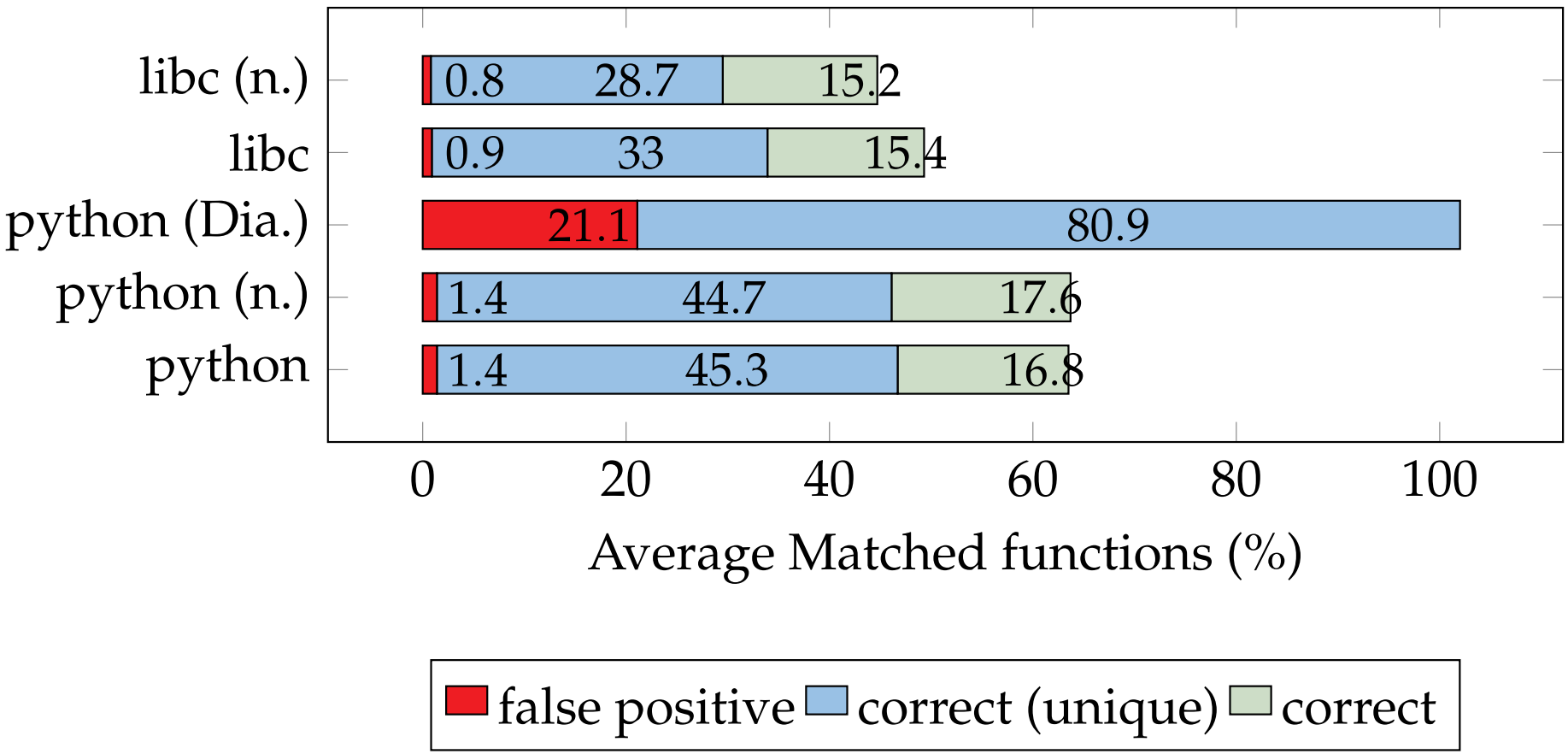


Graph Comparison

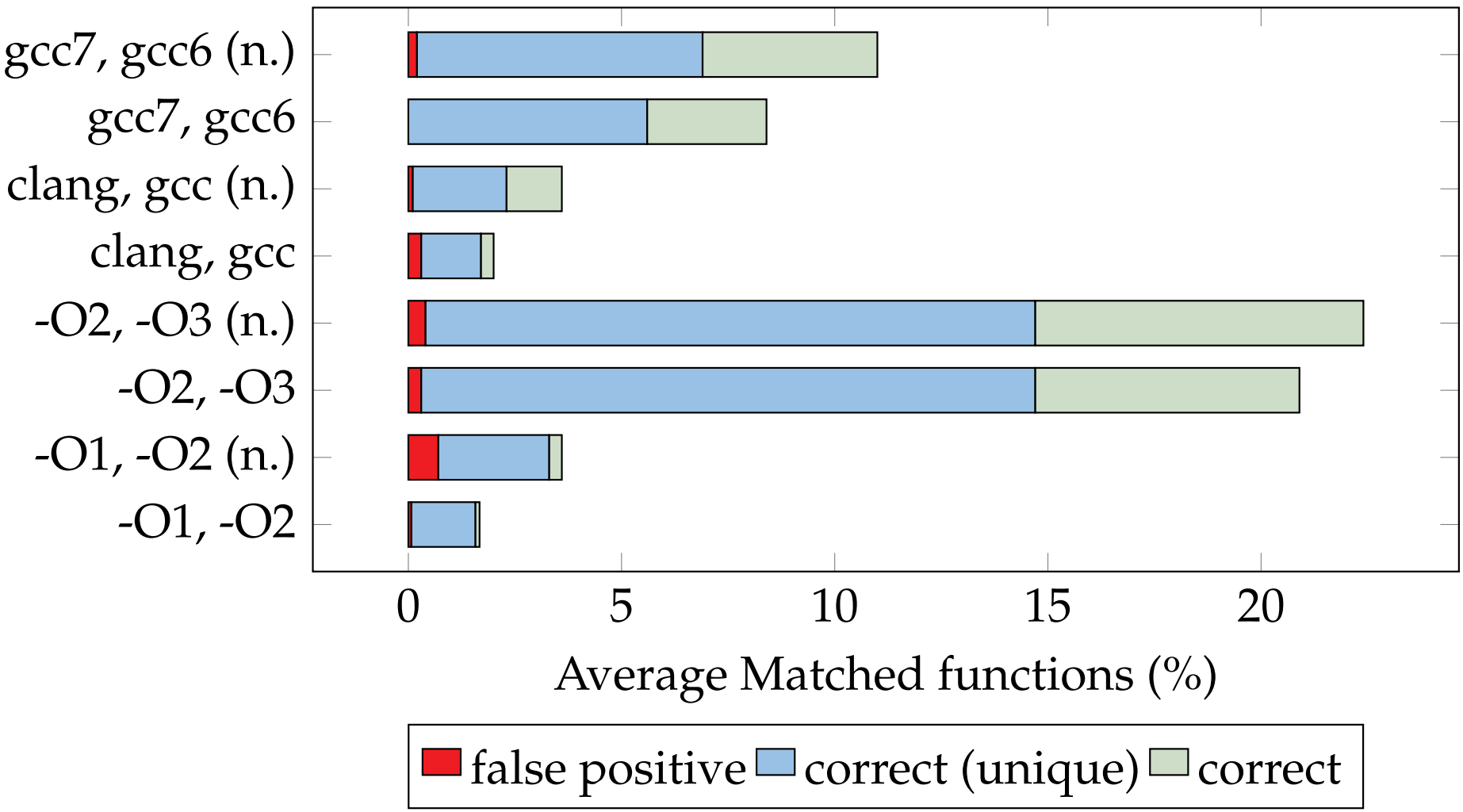




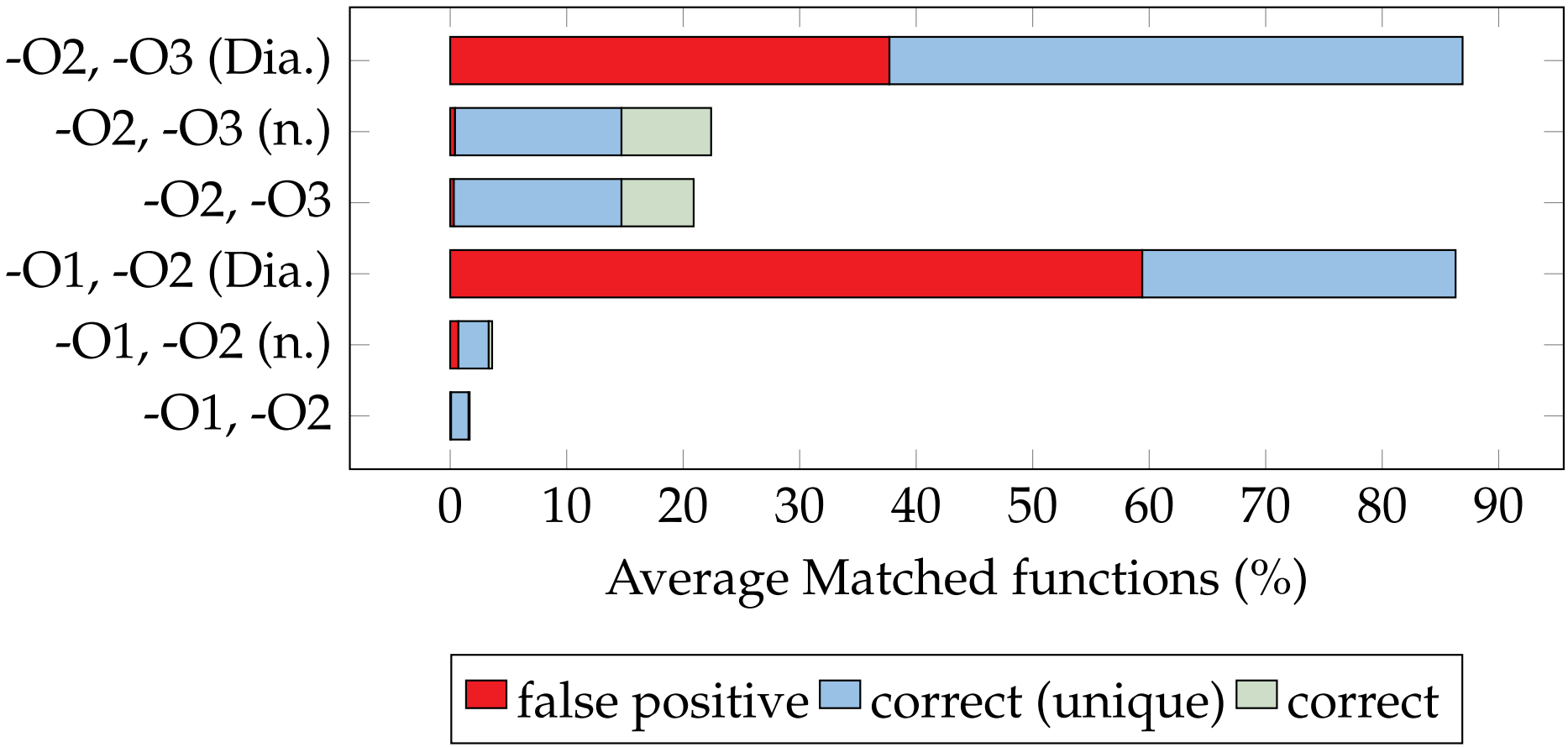
Results


libc (n.)
libc
python (Dia.)
python (n.)
python
0.8
28.7
15.2
0.9
33
15.4
21.1
80.9
1.4
44.7
17.6
1.4
45.3
16.8
0
20
40
60
80
100
Average Matched functions (%)
false positive
correct (unique)
correct

Results


gcc7, gcc6 (n.)
gcc7, gcc6
clang, gcc (n.)
clang, gcc
-O2, -O3 (n.)
-O2, -O3
-O1, -O2 (n.)
-O1, -O2
0
5
10
15
20
Average Matched functions (%)
false positive
correct (unique)
correct

Results


-O2, -O3 (Dia.)
-O2, -O3 (n.)
-O2, -O3
-O1, -O2 (Dia.)
-O1, -O2 (n.)
-O1, -O2
0
10
20
30
40
50
60
70
80
90
Average Matched functions (%)
false positive
correct (unique)
correct

Thanks

jonischk [at] sec.in.tum.de `https://github.com/leetonidas/signatures`