

t1 := Add 1, 1

a := Assign t1

IfGoto 1, .BB1, .BB2 *BB0*

t1 := Assign 2

a := Assign t1

Goto .BB1

BB0

t1 := Assign 2

a := Assign 2

Goto .BB1



BB0

Äquivalenzmengen

`x := Assign 2`

`y := Assign x`

`x := Add x, y`

`t0 := Assign x`

`t1 := Call f, t0, y`

`x := Add y, t1`

Äquivalenzmengen

← []

`x := Assign 2`

`y := Assign x`

`x := Add x, y`

`t0 := Assign x`

`t1 := Call f, t0, y`

`x := Add y, t1`

Äquivalenzmengen

`x := Assign 2`

← `[]`

`y := Assign x`

← `[{x, 2}]`

`x := Add x, y`

`t0 := Assign x`

`t1 := Call f, t0, y`

`x := Add y, t1`

Äquivalenzmengen

`x := Assign 2`

← `[]`

`y := Assign 2`

← `[{x, 2}]`

`x := Add x, y`

← `[{x, y, 2}]`

`t0 := Assign x`

`t1 := Call f, t0, y`

`x := Add y, t1`

Äquivalenzmengen

`x := Assign 2`

← `[]`

`y := Assign 2`

← `[{x, 2}]`

`x := Add 2, 2`

← `[{x, y, 2}]`

`t0 := Assign x`

← `[{y, 2}]`

`t1 := Call f, t0, y`

`x := Add y, t1`

Äquivalenzmengen

`x := Assign 2`

← `[]`

`y := Assign 2`

← `[{x, 2}]`

`x := Add 2, 2`

← `[{x, y, 2}]`

`t0 := Assign x`

← `[{y, 2}]`

`t1 := Call f, t0, y`

← `[{y, 2}, {x, t0}]`

`x := Add y, t1`

Äquivalenzmengen

`x := Assign 2`

← []

`y := Assign 2`

← [{x, 2}]

`x := Add 2, 2`

← [{x, y, 2}]

`t0 := Assign x`

← [{y, 2}]

`t1 := Call f, x, 2`

← [{y, 2}, {x, t0}]

`x := Add y, t1`

Äquivalenzmengen

`x := Assign 2`

← `[]`

`y := Assign 2`

← `[{x, 2}]`

`x := Add 2, 2`

← `[{x, y, 2}]`

`t0 := Assign x`

← `[{y, 2}]`

`t1 := Call f, x, 2`

← `[{y, 2}, {x, t0}]`

`x := Add y, t1`

← Äquivalenzen noch intakt?

$[\{y, 2\}]$



```
t1 := Call func, ptr, 2
```



?

$[\{y, 2\}]$



```
*ptr := Store 2
```



?