

Semantic Stack

$a * := b + c ;$

e.g. Statement Processing

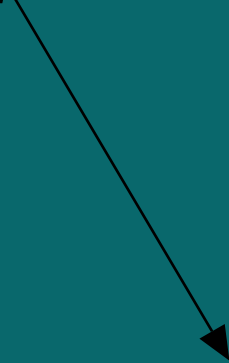
Stacktop



Stacktop



Stacktop



expr(var)
(1,4,False)

Level 1 (global)
Offset 4 rel R15
Indirect False

Stacktop

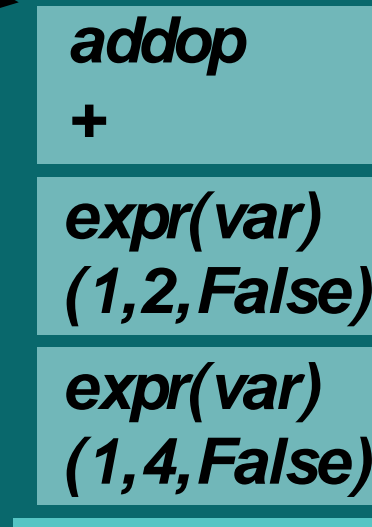


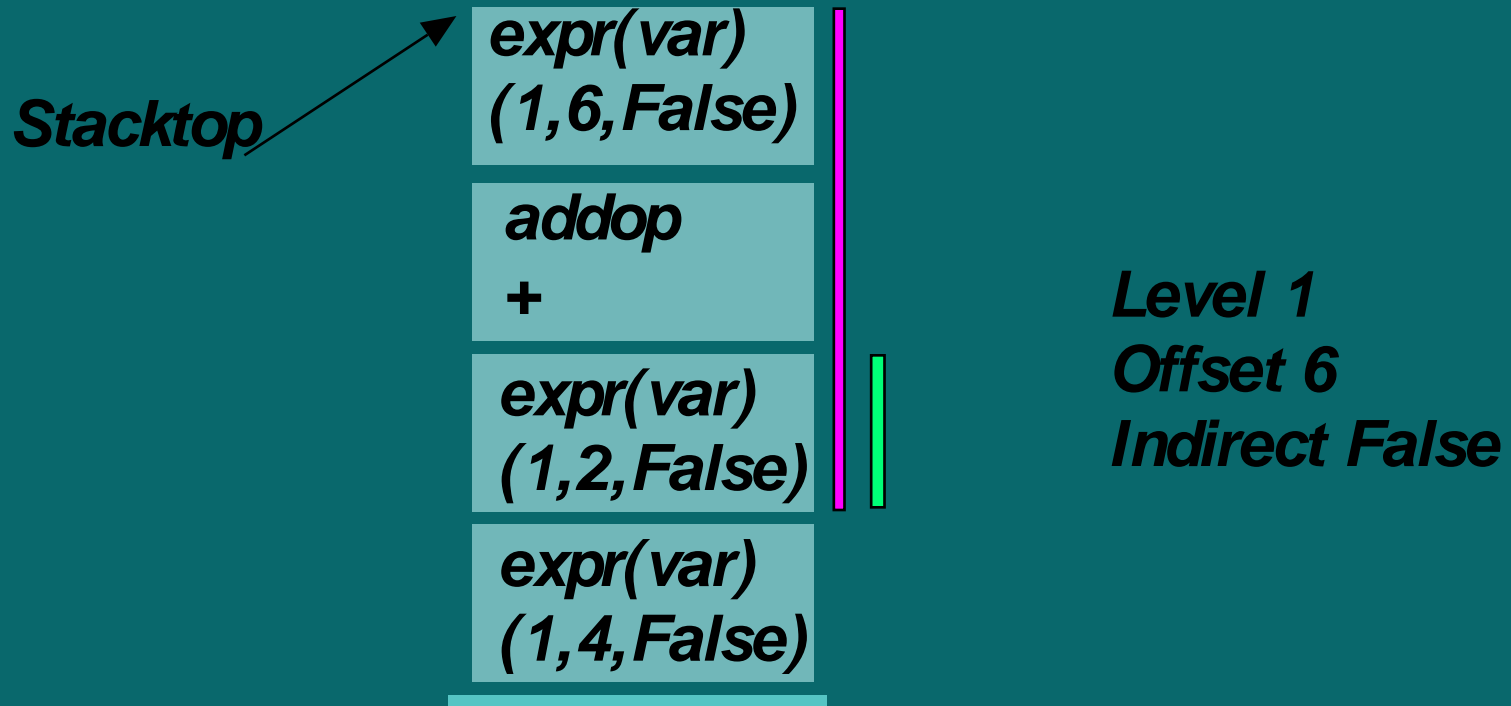
expr(var)
(1,2,False)

expr(var)
(1,4,False)

Level 1
Offset 2
Indirect False

Stacktop





GenAddExpr

Codefile

LoadReg(ST-2)

LD R0, +2(R15)

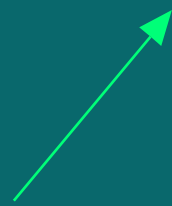
buildOp(st...)
 (indxd, 15, 6)
gen2Address(ia...)

IA R0, +6(R15)

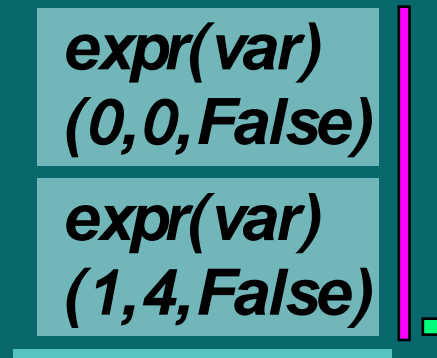
Result

ST := ST-2

...



Stacktop



Level 0 = reg
Offset 0 = R0
Indirect False

GenStore

Codefile

LoadReg(ST)

(no code)

buildOp(st-1...)
 (indxd, 15, 4)

gen2Address(st...)

STO R0, +4(R15)

ST := ST-2

Semantic Stack

a := b + c ;

```
LD R0, +2(R15)
IA R0, +6(R15)
STO R0, +4(R15)
```

Stacktop



Semantic Stack

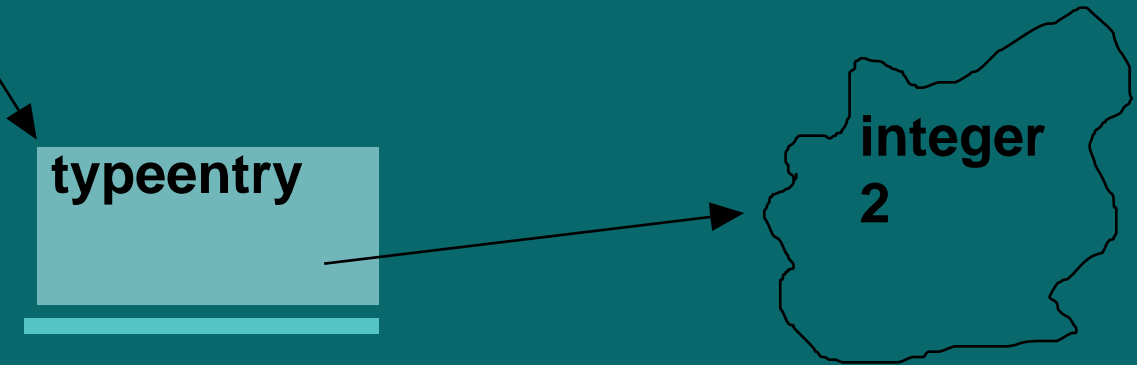
*integer * x, y, z ;*

e.g. Declaration Processing

Stacktop

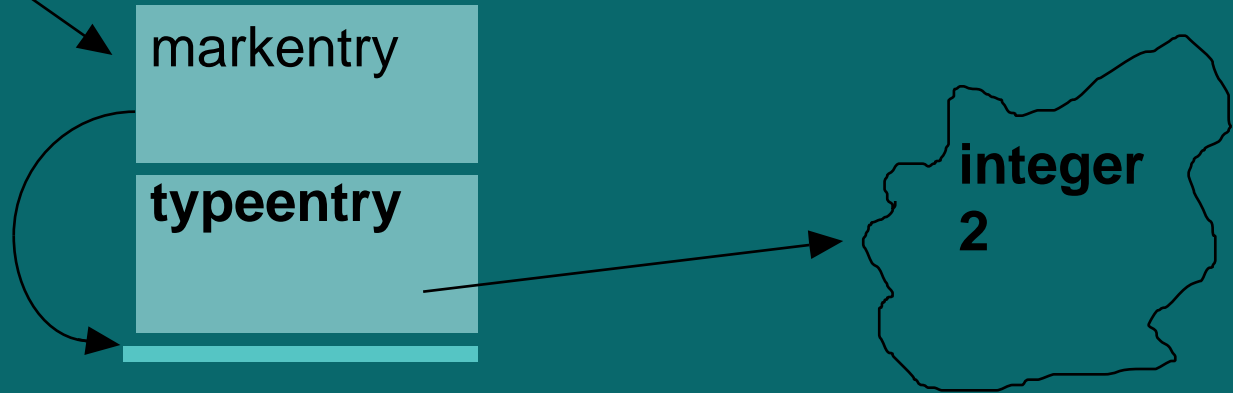


Stacktop



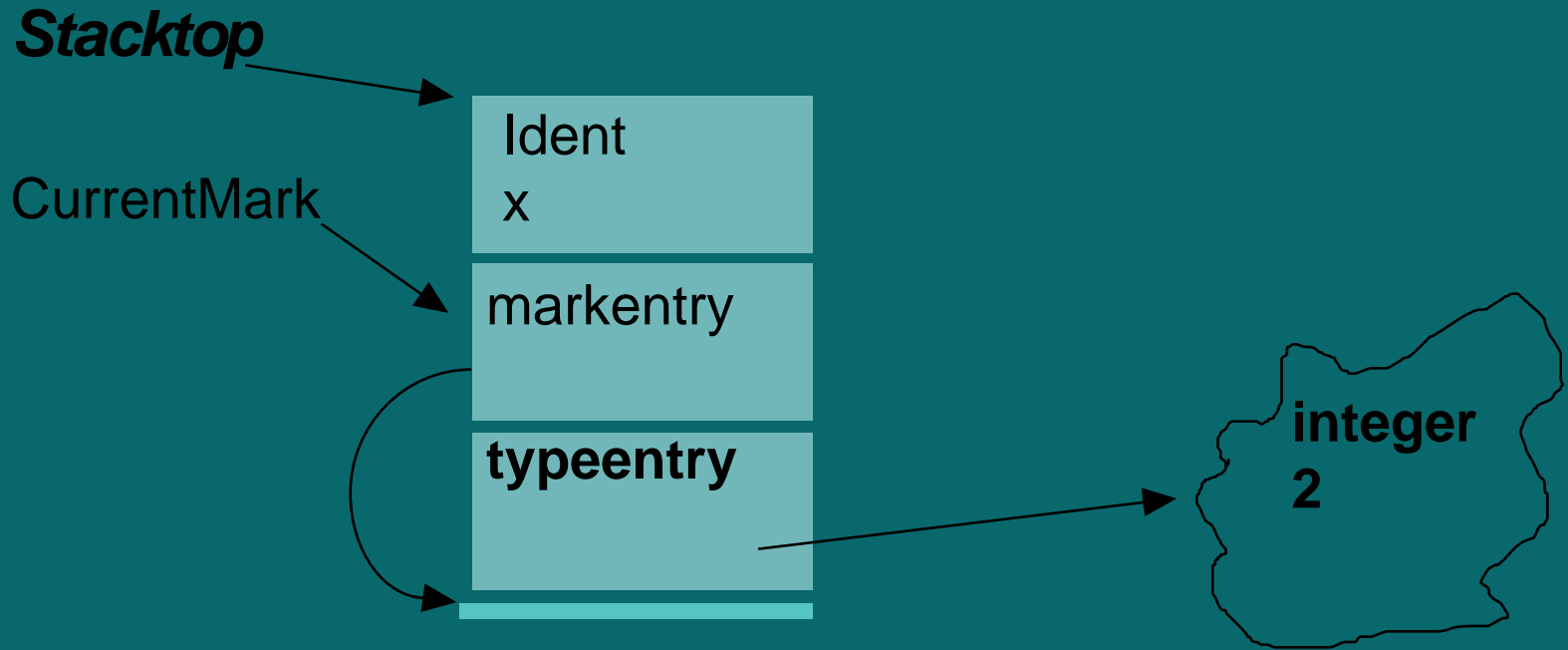
Stacktop

CurrentMark



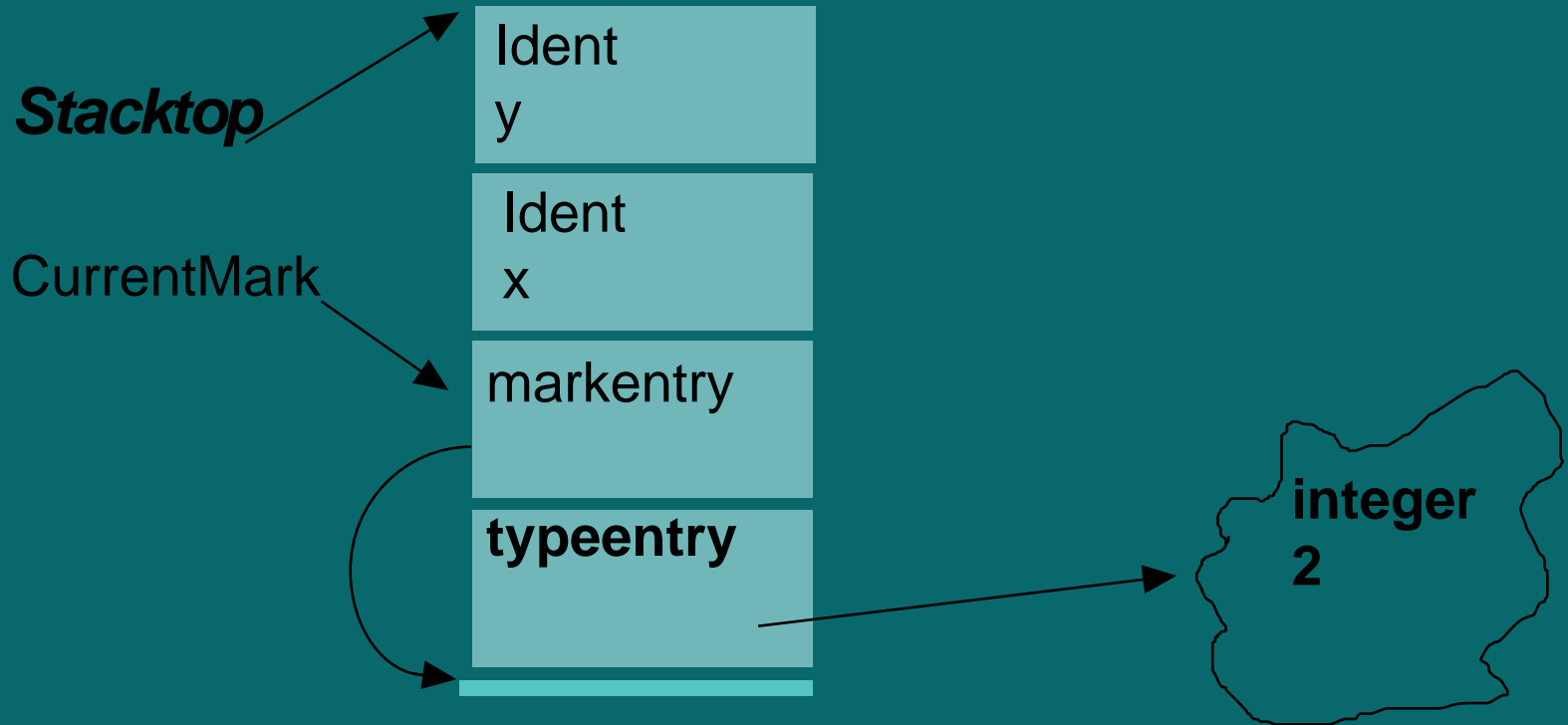
Semantic Stack

*integer x, y, *z;*



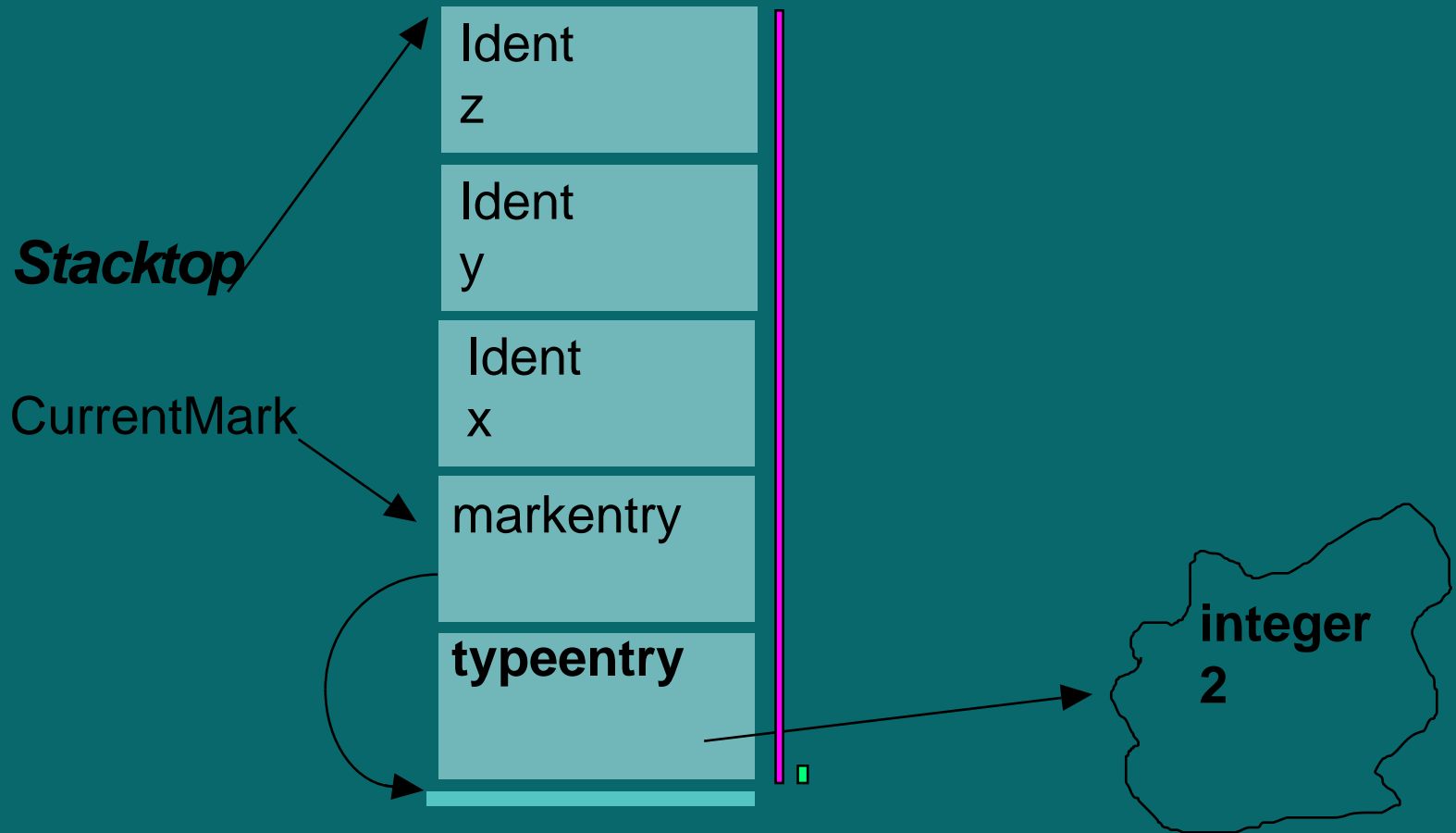
Semantic Stack

*integer x, y, z *;*



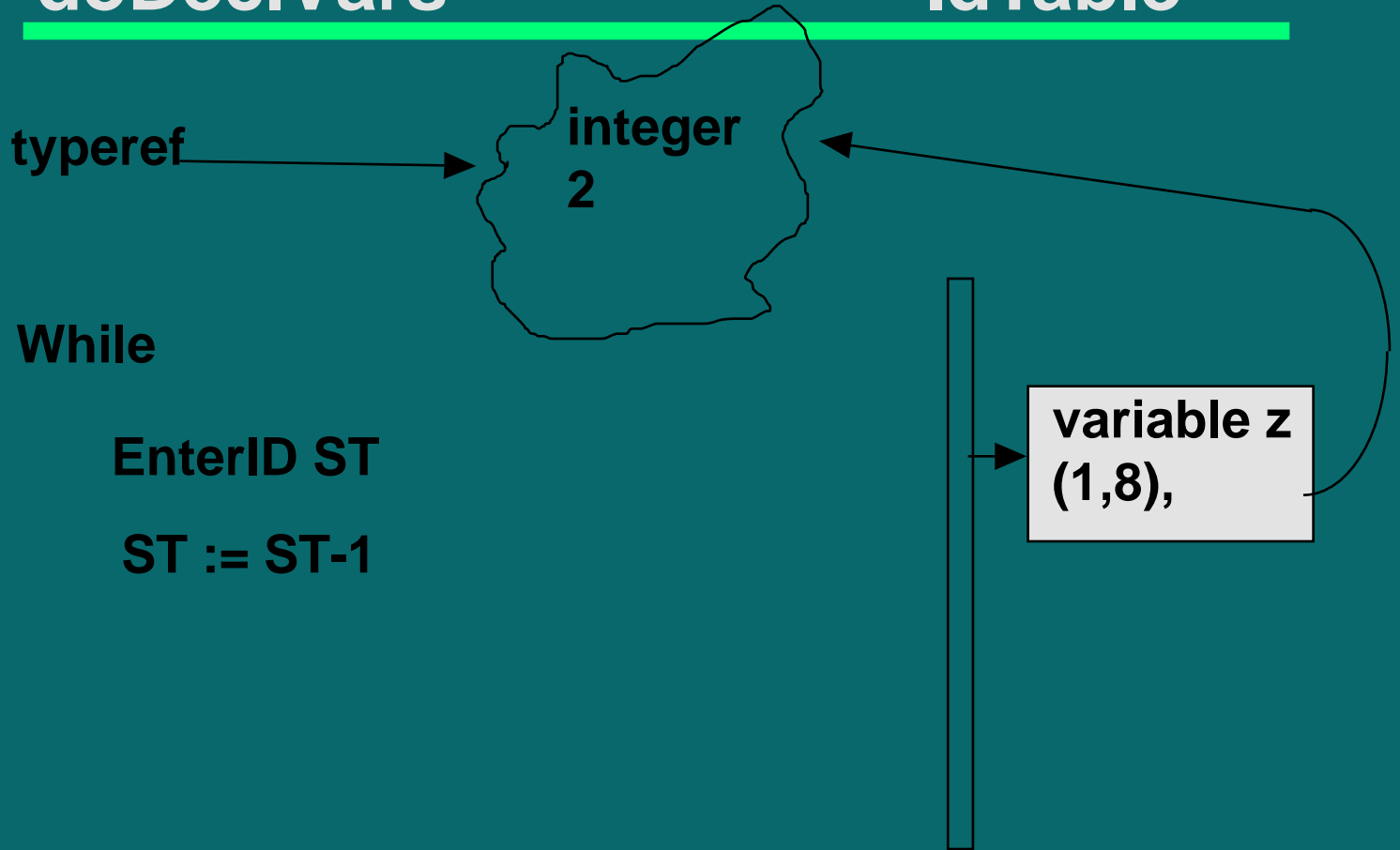
Semantic Stack

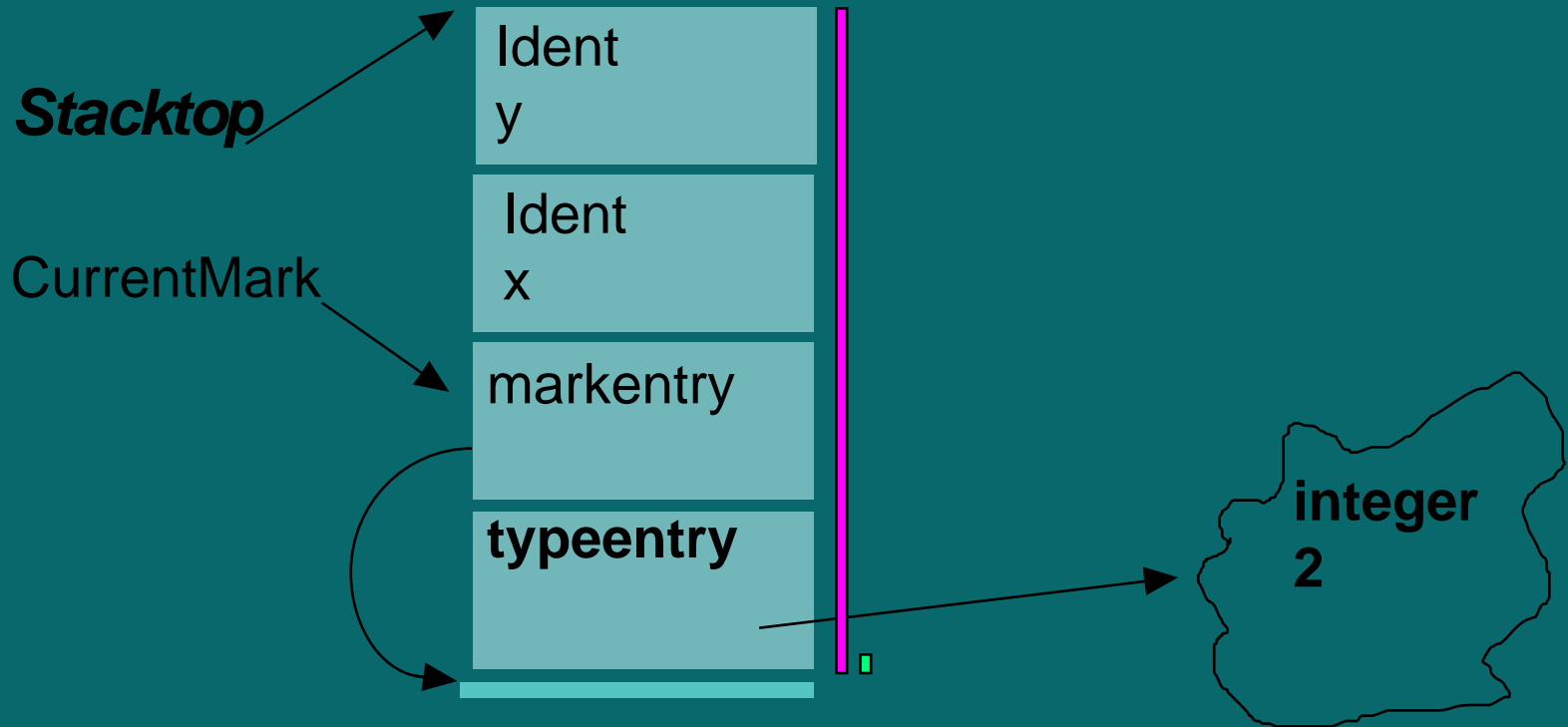
*integer x, y, z *;*



doDeclVars

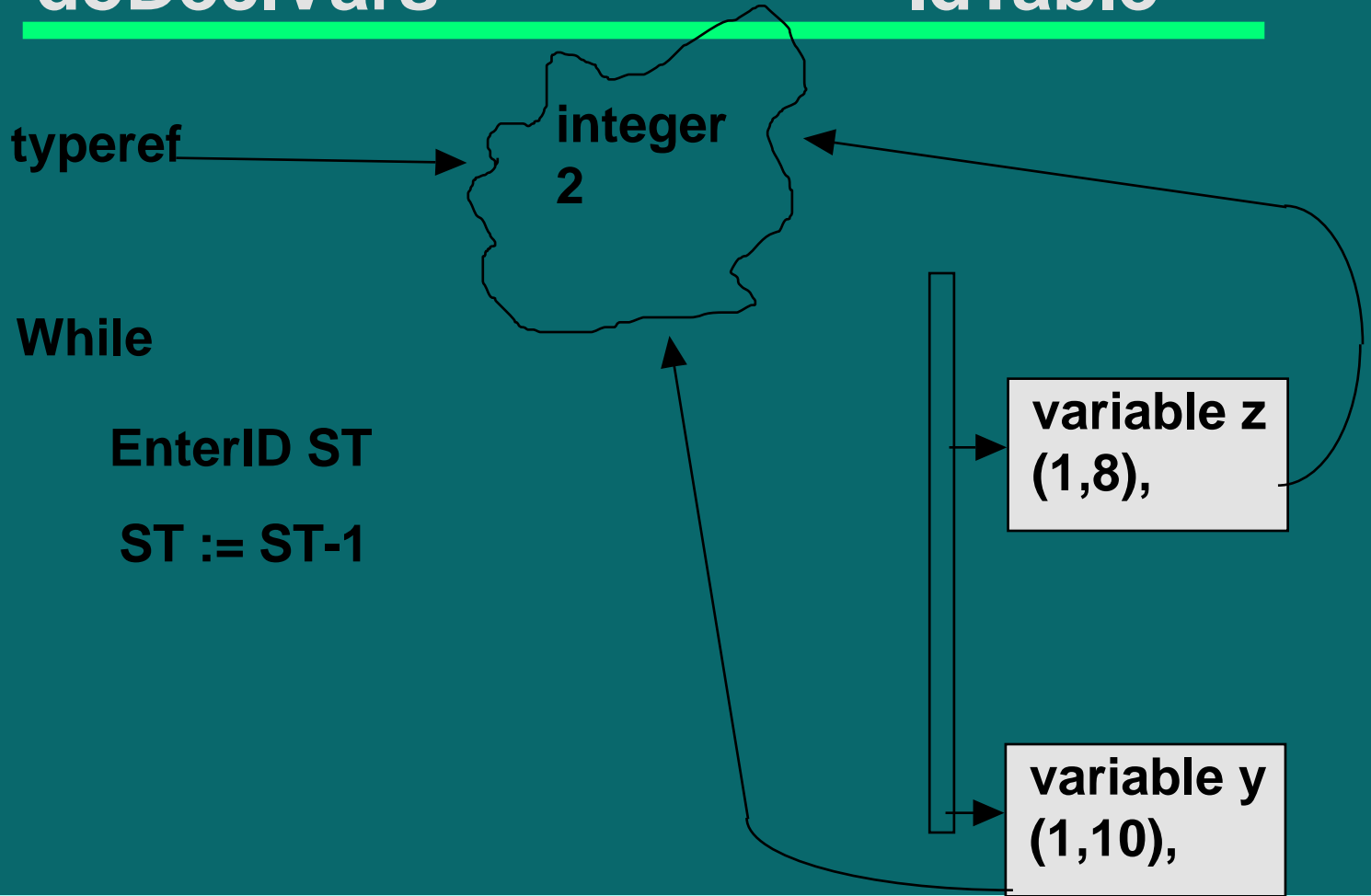
IdTable





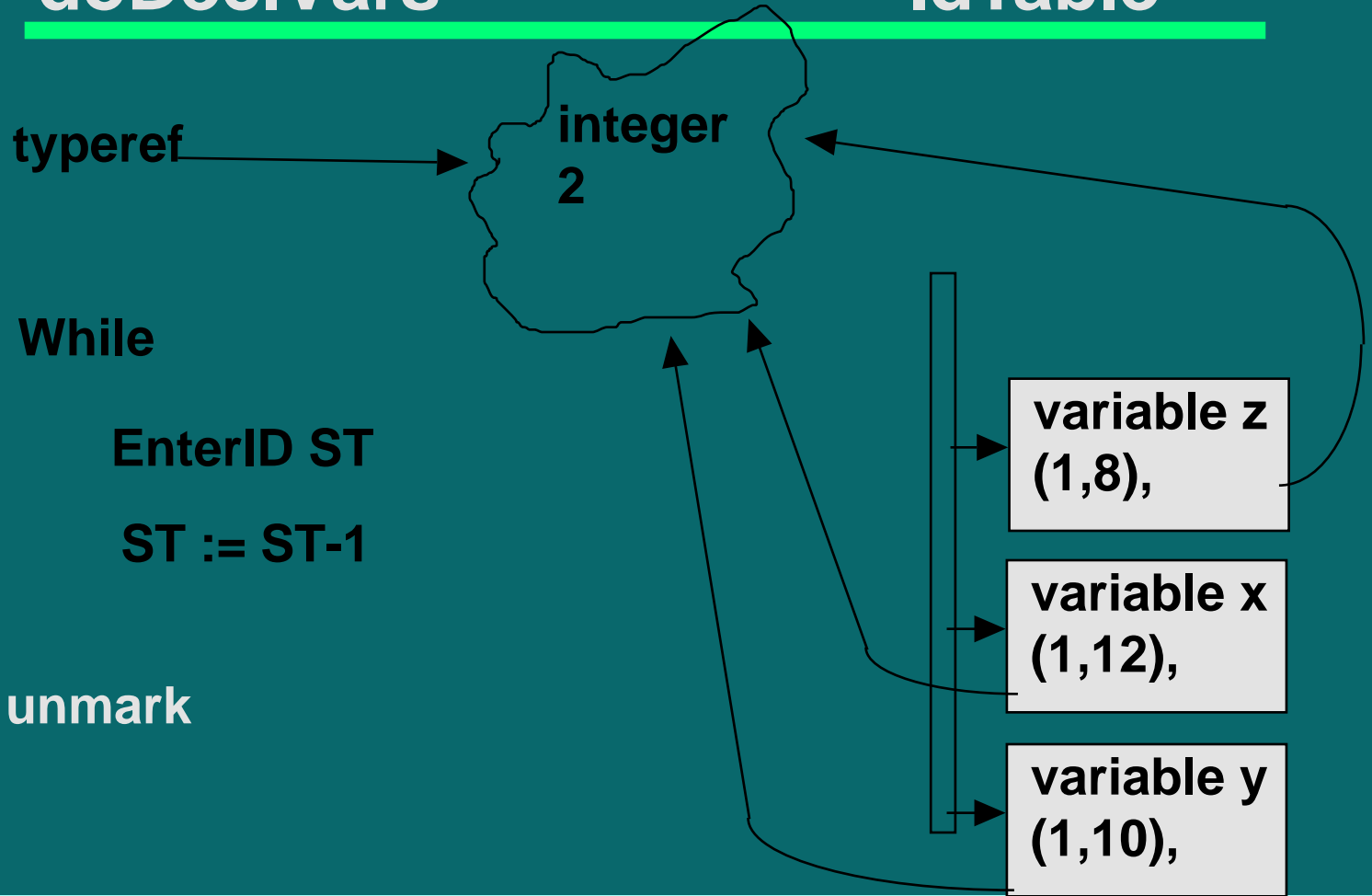
doDeclVars

IdTable



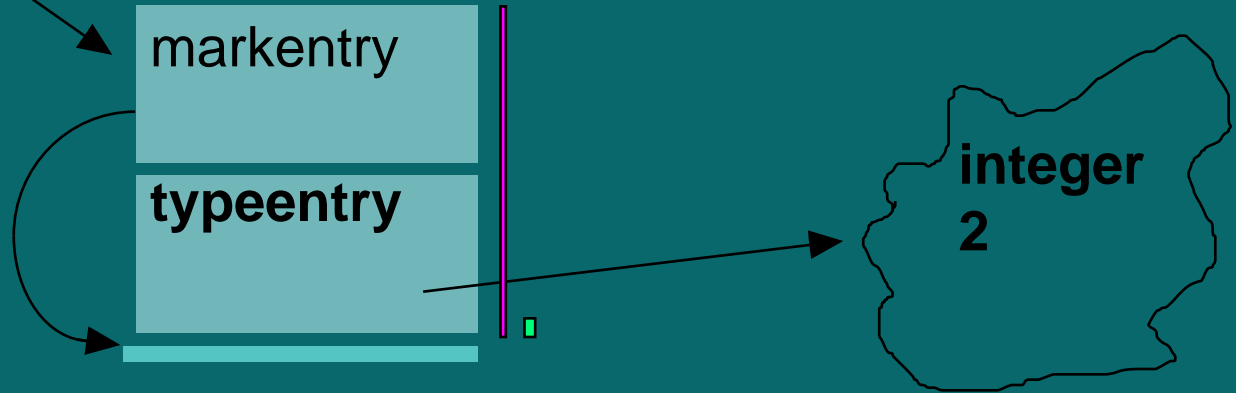
doDeclVars

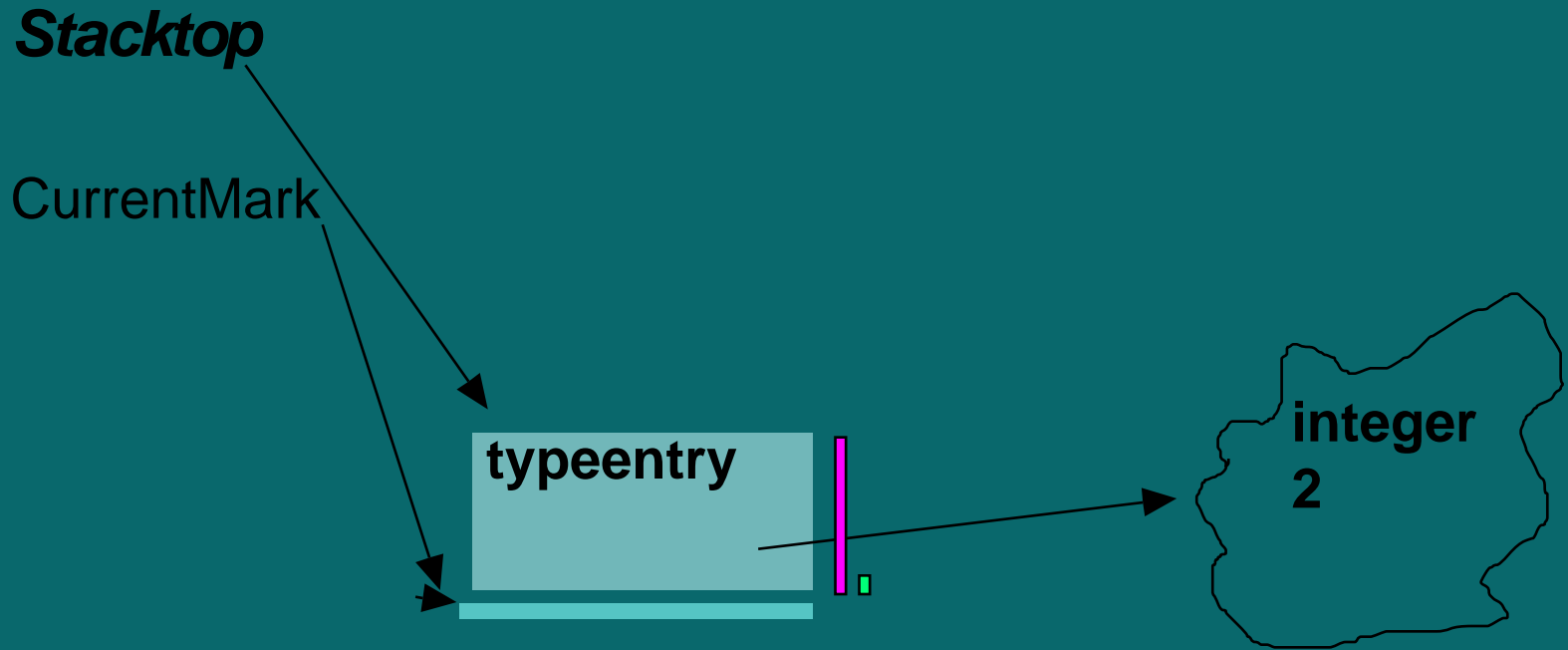
IdTable



Stacktop

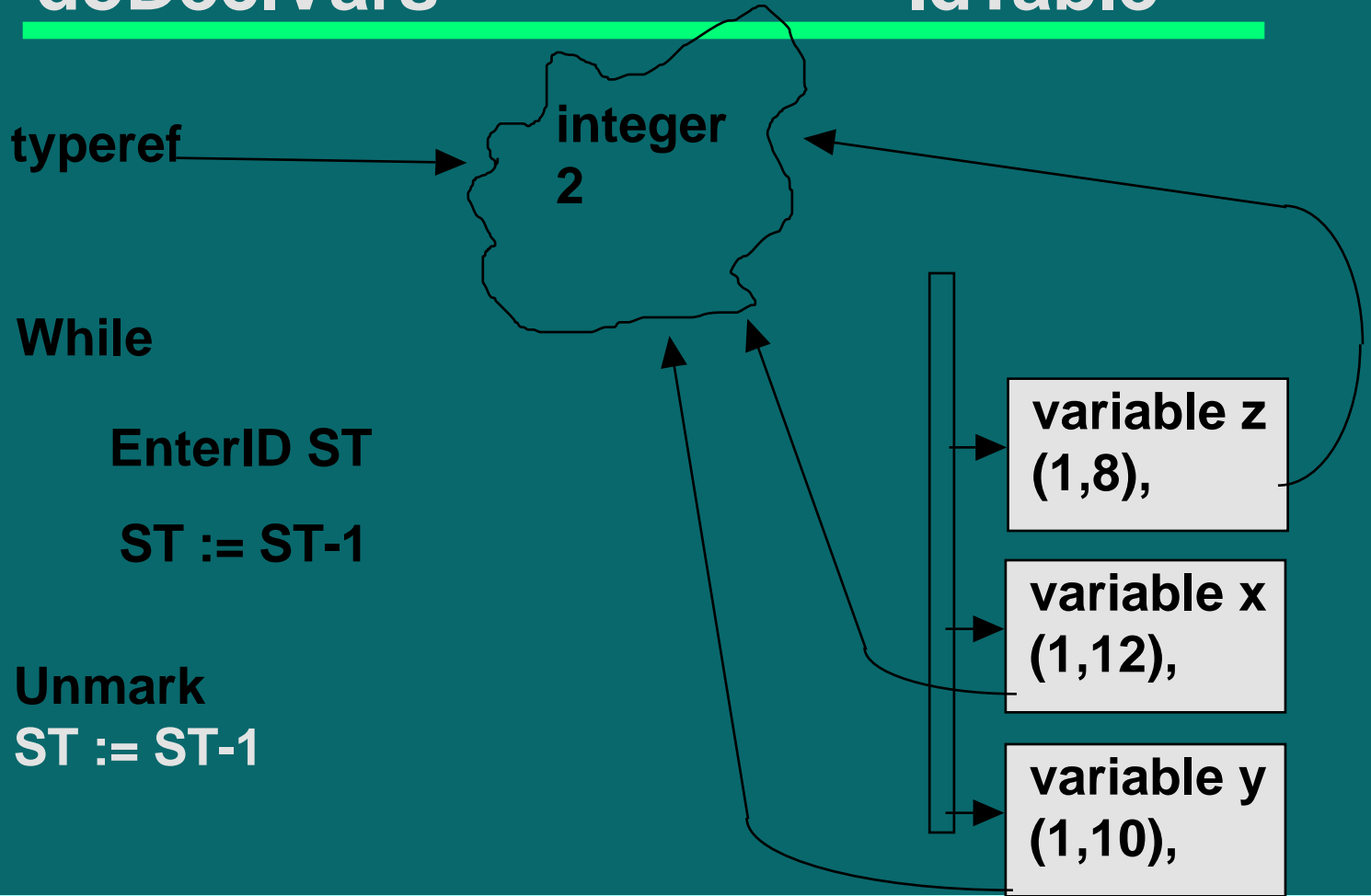
CurrentMark





doDeclVars

IdTable



Semantic Stack

*integer x, y, z ;**

