

Syntaxe abstraite

Tristan Crolard

Laboratoire CEDRIC
Equipe « Systèmes Sûrs »

`tristan.crolard@cnam.fr`

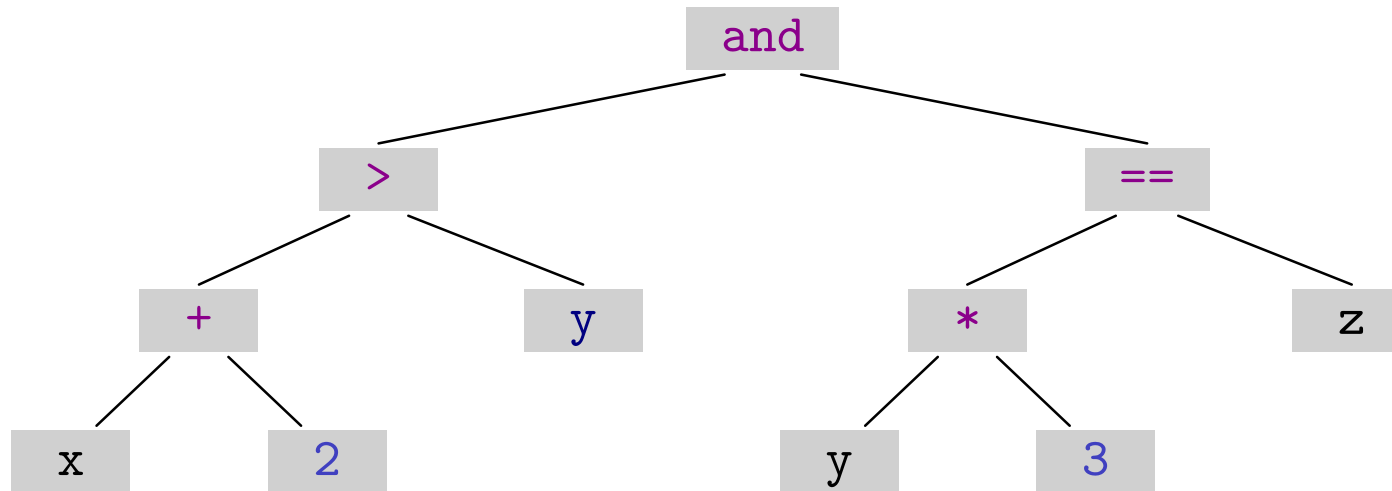
`cedric.cnam.fr/sys/crolard`

Expression arithmétique et booléenne – exemple

Syntaxe concrète

$(x + 2) > y \text{ and } y * 3 == z$

Syntaxe abstraite



Grammaire

$e ::= i$ (integer)
| b (boolean)
| v (variable)
| $e_1 + e_2$
| $e_1 - e_2$
| $e_1 * e_2$
| $e_1 < e_2$
| $e_1 > e_2$
| $e_1 == e_2$
| e_1 and e_2
| e_1 of e_2
| not e_1

Type de données

```
type exp = IntCst    # i: int
          | BoolCst  # b: bool
          | Var      # v: str
          | Plus     # e1: exp; e2: exp
          | Times    # e1: exp; e2: exp
          | Minus    # e1: exp; e2: exp
          | Equal    # e1: exp; e2: exp
          | Less     # e1: exp; e2: exp
          | Greater  # e1: exp; e2: exp
          | And      # e1: exp; e2: exp
          | Or       # e1: exp; e2: exp
          | Not      # e1: exp
```

Type de données – revisité

```
type exp = Union[
  IntCst, # value: int
  BoolCst, # value: bool
  Var, # id: str
  Plus, # left: exp; right: exp
  Times, # left: exp; right: exp
  Minus, # left: exp; right: exp
  Equal, # left: exp; right: exp
  Less, # left: exp; right: exp
  Greater, # left: exp; right: exp
  And, # left: exp; right: exp
  Or, # left: exp; right: exp
  Not, # operand: exp
]
```

Type de données – dataclasses

```
@dataclass
```

```
class IntCst: value: int
```

```
@dataclass
```

```
class BoolCst: value: bool
```

```
@dataclass
```

```
class Var: id: str
```

```
@dataclass
```

```
class Plus: left: exp; right: exp
```

```
@dataclass
```

```
class Times: left: exp; right: exp
```

```
@dataclass
```

```
class Minus: left: exp; right: exp
```

```
@dataclass
class Equal: left: exp; right: exp
```

```
@dataclass
class Less: left: exp; right: exp
```

```
@dataclass
class Greater: left: exp; right: exp
```

```
@dataclass
class And: left: exp; right: exp
```

```
@dataclass
class Or: left: exp; right: exp
```

```
@dataclass
class Not: operand: exp
```

Exemple – revisité

Syntaxe concrète

$(x + 2) > y$ and $y * 3 == z$

Syntaxe abstraite

