

# Programmation Fonctionnelle et Introduction aux Types Abstraits

Langage : OCaml

Resp. UE : Érik Martin-Dorel ([erik.martin-dorel@irit.fr](mailto:erik.martin-dorel@irit.fr))

Université Toulouse III - Paul Sabatier

Année universitaire 2020-2021  
L3 Informatique

# Plan

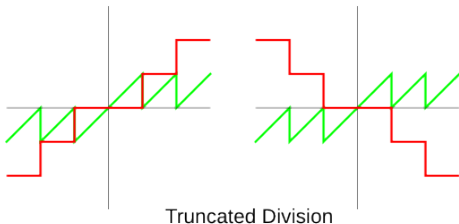
## 1 Autour de l'opérateur infixe "mod"

# Retour sur le modulo et les opérateurs infixes

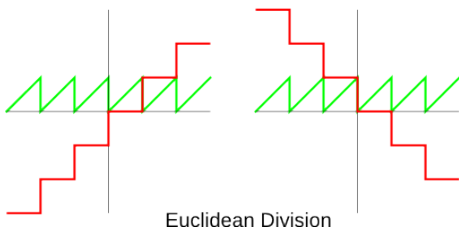
[en.wikipedia.org/wiki/File:Divmod.svg](http://en.wikipedia.org/wiki/File:Divmod.svg)

Positive Divisor

Negative Divisor



Truncated Division



Euclidean Division

Il existe au moins 3 conventions alternatives pour définir le modulo :

[https://en.wikipedia.org/wiki/Modulo\\_operation](https://en.wikipedia.org/wiki/Modulo_operation)

En OCaml «  $x \bmod y$  » est défini comme «  $x - y * (x / y)$  » où «  $x / y$  » est la division tronquée.

Pour avoir un modulo toujours positif, on peut p.ex. définir un infixe

```
let ( % ) x y =
  let m = x mod y in
  if m >= 0 then m
  else m + abs y;;
```