

# Funcons-beta: Trees \*

The P<sub>L</sub>anCompS Project

Trees.cbs | PLAIN | PRETTY

---

## Trees

```
[ Datatype trees
  Funcon tree
  Funcon tree-root-value
  Funcon tree-branch-sequence
  Funcon single-branching-sequence
  Funcon forest-root-value-sequence
  Funcon forest-branch-sequence
  Funcon forest-value-sequence ]
```

Meta-variables  $T <: \text{values}$

Datatype  $\text{trees}(T) ::= \text{tree}(\_ : T, \_ : (\text{trees}(T))^*)$

$\text{trees}(T)$  consists of finitely-branching trees with elements of type  $T$ . When  $V : T$ ,  $\text{tree}(V)$  is a leaf, and  $\text{tree}(V, B_1, \dots, B_n)$  is a tree with branches  $B_1, \dots, B_n$ .

Funcon  $\text{tree-root-value}(\_ : \text{trees}(T)) : \Rightarrow (T)?$

Rule  $\text{tree-root-value tree}(V : T, \_ : (\text{trees}(T))^*) \rightsquigarrow V$

Funcon  $\text{tree-branch-sequence}(\_ : \text{trees}(T)) : \Rightarrow (\text{trees}(T))^*$

Rule  $\text{tree-branch-sequence tree}(\_ : T, B^* : (\text{trees}(T))^*) \rightsquigarrow B^*$

Funcon  $\text{single-branching-sequence}(\_ : \text{trees}(T)) : \Rightarrow T^+$

$\text{single-branching-sequence } B$  extracts the values in  $B$  starting from the root, provided that  $B$  is at most single-branching; otherwise it fails.

Rule  $\text{single-branching-sequence tree}(V : T) \rightsquigarrow V$

Rule  $\text{single-branching-sequence tree}(V : T, B : \text{trees}(T)) \rightsquigarrow$   
 $\text{left-to-right}(V, \text{single-branching-sequence } B)$

Rule  $\text{single-branching-sequence tree}(\_ : T, \_ : \text{trees}(T), \_ : (\text{trees}(T))^+) \rightsquigarrow \text{fail}$

A sequence of trees corresponds to a forest, and the selector funcons on trees  $B$  extend to forests  $B^*$ :

---

\*Suggestions for improvement: [plancomps@gmail.com](mailto:plancomps@gmail.com).  
Reports of issues: <https://github.com/plancomps/CBS-beta/issues>.

*Funcon*  $\text{forest-root-value-sequence}(\_ : (\text{trees}(T))^*) : \Rightarrow T^*$   
*Rule*  $\text{forest-root-value-sequence}(B : \text{trees}(T), B^* : (\text{trees}(T))^*) \rightsquigarrow$   
 $(\text{tree-root-value } B, \text{forest-root-value-sequence } B^*)$   
*Rule*  $\text{forest-root-value-sequence}(\_) \rightsquigarrow (\_)$

*Funcon*  $\text{forest-branch-sequence}(\_ : (\text{trees}(T))^*) : \Rightarrow T^*$   
*Rule*  $\text{forest-branch-sequence}(B : \text{trees}(T), B^* : (\text{trees}(T))^*) \rightsquigarrow$   
 $(\text{tree-branch-sequence } B, \text{forest-branch-sequence } B^*)$   
*Rule*  $\text{forest-branch-sequence}(\_) \rightsquigarrow (\_)$

*Funcon*  $\text{forest-value-sequence}(\_ : (\text{trees}(T))^*) : \Rightarrow T^*$

$\text{forest-value-sequence } B^*$  provides the values from a left-to-right pre-order depth-first traversal.

*Rule*  $\text{forest-value-sequence}(\text{tree}(V : T, B_1^* : (\text{trees}(T))^*), B_2^* : (\text{trees}(T))^*) \rightsquigarrow$   
 $(V, \text{forest-value-sequence } B_1^*, \text{forest-value-sequence } B_2^*)$   
*Rule*  $\text{forest-value-sequence}(\_) \rightsquigarrow (\_)$

Other linearizations of trees can be added: breadth-first, right-to-left, C3, etc.