

# MANUAL FOR THE STRUCTURE BINARYTREETWITHROOTANDTOPS

## 1. STRUCTURE BINARYTREETWITHROOTANDTOPS

1.1. **Notation.** Given a Pair  $p = (p.1, p.2)$ , we have

- $\text{Finset } \alpha$  denotes a *finite* subset of  $\alpha$  (Lean’s `Finset`).
- Given  $p : \text{Finset } \alpha \times \text{Finset } \alpha$ ,  $\text{pairToFinset}(p) = \{p.1, p.2\}$ .

```
noncomputable def pairToFinset {α : Type*} [DecidableEq α] : α × α → Finset α
| (p1, p2) => {p1, p2}
```

- $\text{Combinatorial\_Support}(p) = p.1 \cup p.2$  is the “combinatorial support”

```
noncomputable def Combinatorial_Support {α : Type*} [DecidableEq α] (p : Finset α × Finset α) :
Finset α := p.1 ∪ p.2
```

- $\text{Disjoint}(A, B)$  abbreviates  $A \cap B = \emptyset$ .

1.2. **Overview. Data.** The record

$\text{BinaryTreeWithRootandTops}(\alpha : \text{Type}^*) \quad [\text{DecidableEq } \alpha]$

stores four finite objects:

- **Childs** :  $\text{Finset } \alpha$  — the ground set (assumed non-empty);
- **Root** :  $\text{Finset } \alpha \times \text{Finset } \alpha$  — the first split of **Childs**;
- **Branches** :  $\text{Finset}(\text{Finset } \alpha \times \text{Finset } \alpha)$  — all interior nodes, including **Root**;
- **Tops** :  $\text{Finset } \alpha$  — the labels of distinguished leaves.

Laminar family of supports. The supports of the branches satisfy two crucial axioms:

(L1) *Cover.*  $\bigcup_{p \in \text{Branches}} \text{Combinatorial\_Support}(p) = \text{Childs}.$

(L2) *Nesting.* Whenever  $p \neq q$  in **Branches**, the supports are either disjoint or properly nested:

$$\begin{aligned} \text{Combinatorial\_Support}(p) \cap \text{Combinatorial\_Support}(q) &= \emptyset \quad \text{or} \\ \text{Combinatorial\_Support}(p) &\subseteq \text{Combinatorial\_Support}(q) \quad \text{or} \\ \text{Combinatorial\_Support}(q) &\subseteq \text{Combinatorial\_Support}(p). \end{aligned}$$

Indeed something even stronger happens, the family

$$\mathcal{L} = \{p.1, p.2 \mid p \in \text{Branches}\}$$

is a *maximal laminar family* (nested set system) on **Childs**. The record not only remembers this hierarchy of blocks but also keeps track of *how each block is split*: for every non-leaf support  $S \in \mathcal{L}$  there exists *at most one* ordered pair  $(A, B) \in \text{Branches}$  with  $A \cup B = S$ ,  $A \cap B = \emptyset$  and  $\text{Combinatorial\_Support}(A, B) = S$ . This extra orientation (left vs. right child) upgrades the laminar family into a full binary tree.

Singleton supports  $\{t\}$  correspond precisely to the leaf pairs  $(\{t\}, \{\})$  or  $(\{\}, \{t\})$ ; the set of all such labels is stored in the field **Tops**.

### 1.3. Fields.

**Root** :  $\text{Finset}\alpha \times \text{Finset}\alpha$ : The distinguished ordered pair that serves as the root of the tree.

**Childs** :  $\text{Finset}\alpha$ : The finite set of all basic symbols that may appear in either component of a branch pair.

**Branches** :  $\text{Finset}(\text{Finset}\alpha \times \text{Finset}\alpha)$ : The finite family of ordered pairs that form the vertices of the tree, including **Root**.

**Tops** :  $\text{Finset}\alpha$ : A specified finite subset of **Childs** whose singletons appear as leaf pairs.

**1.4. Structure properties.** All quantifiers range over the corresponding finite sets declared above.

**(1) Non-empty components.**

$$\forall y \in \text{Branches}, y.1 \neq \emptyset \wedge y.2 \neq \emptyset.$$

**(2) Disjoint components.**

$$\forall p \in \text{Branches}, p.1 \cap p.2 = \emptyset.$$

**(3) Distinct pairs are disjoint as sets.**

$$\forall p, q \in \text{Branches}, p \neq q \rightarrow (\text{pairToFinset}(p)) \cap (\text{pairToFinset}(q)) = \emptyset.$$

**(4) Root belongs to the branch set.**

$$\text{Root} \in \text{Branches}.$$

**(5) Every child occurs in some branch.**

$$\forall a \in \text{Childs}, \exists p \in \text{Branches}, a \in \text{Combinatorial\_Support}(p).$$

**(6) Children are contained in the root pair.**

$$\text{Childs} \subseteq \text{Root}.1 \cup \text{Root}.2.$$

**(7) Branch components are made of children.**

$$\forall p \in \text{Branches}, p.1 \subseteq \text{Childs} \wedge p.2 \subseteq \text{Childs}.$$

**(8) Recursive tree structure.**

$$\forall p \in \text{Branches}, p \neq \text{Root} \rightarrow \exists q \in \text{Branches}, p.1 \cup p.2 \in \text{pairToFinset}(q).$$

**(9) Tops are witnessed by singleton pairs.**

$$\forall t \in \text{Tops}, \exists q \in \text{Branches}, \{t\} \in \text{pairToFinset}(q).$$

**(10) Singletons in any branch are tops.**

$$\forall p \in \text{Branches}, \forall x, \{x\} \in \text{pairToFinset}(p) \rightarrow x \in \text{Tops}.$$

**(11) Non-empty root and child set.**

$$\text{Root}.1 \neq \emptyset, \text{Root}.2 \neq \emptyset, \text{Childs} \neq \emptyset.$$

**(12) Root components are disjoint.**

$$\text{Root}.1 \cap \text{Root}.2 = \emptyset.$$

**(13) Support property (tree compatibility).**

$$\begin{aligned} & \forall p, q \in \text{Branches}, p \neq q \rightarrow \\ & (\text{Combinatorial\_Support}(p) \cap \text{Combinatorial\_Support}(q) = \emptyset) \vee \\ & \text{Combinatorial\_Support}(p) \subseteq q.1 \vee \text{Combinatorial\_Support}(p) \subseteq q.2 \vee \\ & \text{Combinatorial\_Support}(q) \subseteq p.1 \vee \text{Combinatorial\_Support}(q) \subseteq p.2. \end{aligned}$$

```

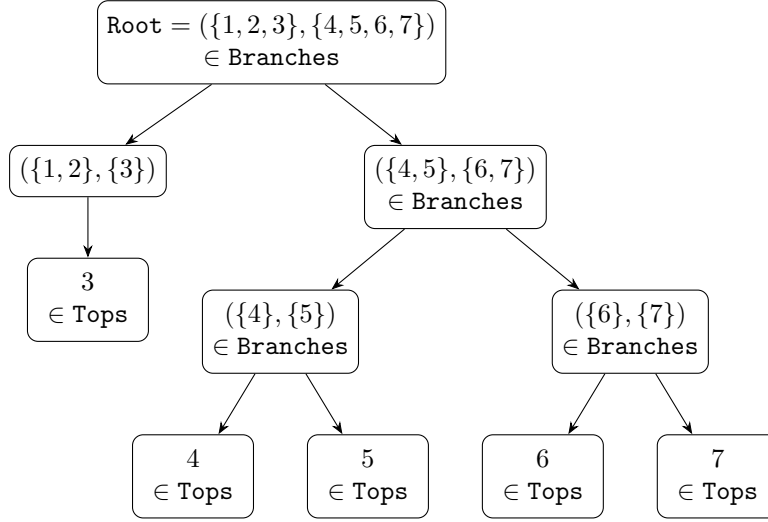
structure BinaryTreeWithRootandTops ( $\alpha$  : Type*) [DecidableEq  $\alpha$ ] where
  Root      : Finset  $\alpha$  × Finset  $\alpha$ 
  Childs    : Finset  $\alpha$ 
  Branches  : Finset (Finset  $\alpha$  × Finset  $\alpha$ )
  NonemptyPairs:
    |  $\forall y \in \text{Branches}, y.1.\text{Nonempty} \wedge y.2.\text{Nonempty}$ 
  Tops: Finset  $\alpha$ 
  DisjointComponents:
    |  $\forall p \in \text{Branches}, \text{Disjoint } p.1 \ p.2$ 
  DistinctComponentsPairs:
    |  $\forall p \in \text{Branches}, \forall q \in \text{Branches}, (p \neq q) \rightarrow \text{Disjoint } (\text{pairToFinset } p) (\text{pairToFinset } q)$ 
  RootinBranches:
    |  $\text{Root} \in \text{Branches}$ 
  EveryChildinaBranch:
    |  $\forall a \in \text{Childs}, \exists p \in \text{Branches}, a \in \text{Combinatorial\_Support } p$ 
  RootcontainsChilds:
    |  $\text{Childs} \subseteq \text{Root}.1 \cup \text{Root}.2$ 
  TreeStructureChilds:
    |  $\forall p \in \text{Branches}, p.1 \subseteq \text{Childs} \wedge p.2 \subseteq \text{Childs}$ 
  TreeStructure:
    |  $\forall p \in \text{Branches}, p \neq \text{Root} \rightarrow \exists q \in \text{Branches}, p.1 \cup p.2 \in \text{pairToFinset } q$ 
  TopsareTops:
    |  $\forall p \in \text{Tops}, \exists q \in \text{Branches}, \{p\} \in \text{pairToFinset } q$ 
  SingletonsAreTops :
    |  $\forall p \in \text{Branches},$ 
    |   |  $\forall x, \{x\} \in \text{pairToFinset } p \rightarrow (x \in \text{Tops})$ 
  NonemptyRoot:  $\text{Root}.1.\text{Nonempty} \wedge \text{Root}.2.\text{Nonempty}$ 
  NonemptyChilds:  $\text{Childs}.\text{Nonempty}$ 
  DisjointRoot:  $\text{Disjoint } \text{Root}.1 \ \text{Root}.2$ 
  SupportProperty :
    |  $\forall p \in \text{Branches}, \forall q \in \text{Branches}, p \neq q \rightarrow$ 
    |   |  $(\text{Disjoint } (\text{Combinatorial\_Support } p) (\text{Combinatorial\_Support } q)) \vee$ 
    |   |  $\text{Combinatorial\_Support } p \subseteq q.1 \vee$ 
    |   |  $\text{Combinatorial\_Support } p \subseteq q.2 \vee$ 
    |   |  $\text{Combinatorial\_Support } q \subseteq p.1 \vee$ 

```

1.5. **Example.** Let

$$\begin{aligned}
 \text{Childs} &= \{1, 2, 3, 4, 5, 6, 7\}, \\
 \text{Root} &= (\{1, 2, 3\}, \{4, 5, 6, 7\}), \\
 \text{Branches} &= \{(\{1, 2, 3\}, \{4, 5, 6, 7\}), (\{1, 2\}, \{3\}), (\{4, 5\}, \{6, 7\}), \\
 &\quad (\{4\}, \{5\}), (\{6\}, \{7\})\}, \\
 \text{Tops} &= \{3, 4, 6, 7\}.
 \end{aligned}$$

and we can see the tree structure graphically below



## MAIN RESULT

**Theorem 1.1** (`exists_tree_childs_eq_C_and_all_childs_in_Tops_of_card_ge_two`). *Let  $C$  be a finite set with at least two elements. Then there exists a structure*

$$T : \text{BinaryTreeWithRootandTops}$$

*such that*

$$T.\text{Childs} = C \quad \text{and} \quad T.\text{Tops} = C.$$

```
theorem exists_tree_childs_eq_C_and_all_childs_in_Tops_of_card_ge_two
{α : Type*} [DecidableEq α] {C : Finset α} (hC : C.card ≥ 2) :
  ∃ T : BinaryTreeWithRootandTops α, T.Childs = C ∧ C = T.Tops := by
```

**1.6. Example.** In the previous example we do not have  $T.\text{Tops} = T.\text{Childs}$ , however we can complete it as

