Biset

## TREE + SEED

Biset TREE + SEED

Tree is a **random sequencer** based on **tree growth and wind simulation**. It can generate sequences up to **64 steps long.** The sequences generated by **Tree** can be influenced by wind force, allowing sequences to evolve continuously while keeping the same dynamics.

Seed is a **Tree expander** allowing to easily create evolving melodies from **Tree** outputs.
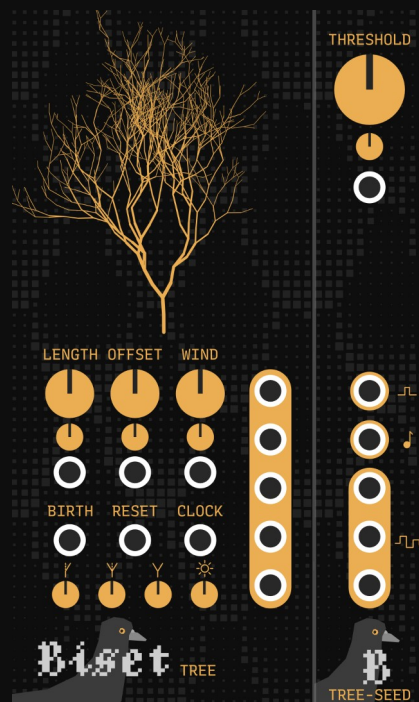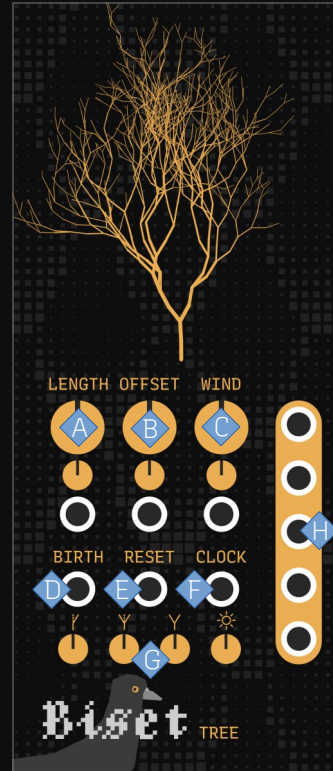
Biset TREE

It's quite easy to create sequences with **Tree**. You simply need to define a **sequence length** (**A**) and provide a **clock input** (**F**). **Tree** will then output a randomly generated sequence for each of it's **5 bipolar outputs** (-5v/+5v).

A set of **5 random values** is assigned to **each branch of the tree**. A sequence is simply made by visiting **neighboring branches** and getting their values. The **offset control** allows you to **scan** the tree from a **different starting point**, thus getting new sequences. The **wind control** allows you to add the wind force currently on the visited branch to its values, thus making the sequence less repetitive.

Wind effect is **stronger on small branches** which means that by **moving the offset control**, you will also potentially change the **wind effect** on the sequence.

A	Sequence length control
B	Sequence offset / scan
C	Wind force on sequence
D	New tree trigger
E	Reset sequence trigger
F	Sequence clock trigger
G	Tree controls
H	Outputs

Tree growth controls (**G**) allows you to define how the tree will grow. These controls will probably only have a negligible effect on the sequences.

Tree also includes a **mutation control** in it's **context menu** (right click). This controls define the chance for each branch to generate new random values each time it is read. With some **wind force**, a **really small chance of mutation** is enough to create really interesting sequences (ex: 2%).

Biset SEED

Seed is made to create an always evolving **melody** from the outputs of it's connected **Tree**.

The **Tree 1st output** is used to define the **melody gate state**. If the value is **greater than Seed threshold** (**A**), the gate will be **on**, otherwise, it will be **off**. The **2nd output** will be used to define the **melody pitch**. The **pitch range** can be set via the **context menu** (right click). The **other outputs** can be used for **CV modulation**.

Gate mode can be set via the **context menu** (gate or trigger).

Polyphony can be set via the **context menu**. The sequence will only output **one note at a time** but polyphony allows the potential envelope generators used with the sequencer to end last note while triggering the next one.

It's important to note that the **Seed threshold control** is **not** a density control. Even at **50%**, the melody can still have more or less than **50%** of the sequence notes active.

THRESHOLD

TREE-SEED

A	Sequence gate threshold
B	Gate / trigger output
C	Pitch output
D	CV outputs