- Initialize $w^k \leftarrow 1$ for $k = 1 : K$
- Execute K copies of program $\mathcal{P}$ *in parallel*, call them $\mathcal{P}^k$
- While executing $\mathcal{P}^k$ if a `sample`, `observe`, or `predict` is reached do:
  - `sample`: $\mathcal{P}^k$ passes to us a continuation $\mathtt{k}^k$ and a tuple (f^k, θ^k) consisting of a distribution f^k and a parameter vector θ^k. We sample a value $x^k \sim f^k(\cdot|\theta^k)$ then call $(\mathtt{k}^k\ \mathtt{x}^k)$ which continues execution of $\mathcal{P}^k$ provided the value.
  - `observe`: wait for all K $\mathcal{P}^k$ to reach `observe`
    - all $\mathcal{P}^k$ pass us a continuation $\mathtt{k}^k$ and a tuple (g^k, ϕ^k, y^k) consisting of a distribution g^k, a parameter vector ϕ^k, and a observed value y^k. We compute $w^k \leftarrow w^k g(y^k|\phi^k)$.
    - subselect K continuations $\mathtt{k}^j$ with probability $\frac{w^j}{\sum_k w^k}$, set each $w^j \leftarrow \frac{1}{K}\sum_k w^k$, and call all $(\mathtt{k}^j)$ *in parallel*.
  - `predict`: $\mathcal{P}^k$ passes us a continuation $\mathtt{k}^k$, a label ℓ^k, and a value z^k. We store (ℓ^k, z^k) and call $(\mathtt{k}^k)$.
- Once all K copies of $\mathcal{P}^k$ reach the end of the program "output" $(\ell^1, z^1, w^1) \ldots (\ell^K, z^K, w^K)$