```
Build-Graph(Env) → Graph
   nodes ← {τ̂ | τ̂ ∈ E}
   edges ← ∅
   to-process ← {τ̂ | τ̂ ∈ E ∧ is-poly(τ̂)}

   while to-process ≠ ∅ loop
      p ← to-process.pop()
      for τ̂ ∈ nodes loop
         if can-pass-to(τ̂, p) then
            edges ← edges ∪ {(p, p τ̂)}
            nodes ← nodes ∪ {p τ̂}
            if is-poly(p τ̂) then
               to-process = to-process ∪ {p τ̂}

   return (nodes, edges)
```

```
fx(τ̂, E) → ε
   graph ← Build-Graph(E)
   subgraph ← nodes reachable from τ̂ in graph
   return {fx(τ̂′) | τ̂′ ∈ subgraph ∧ ¬is-poly(τ̂′)}
```