

---

**Algorithm 1** Débinarisation

---

```
function BINAIRE2NAIRE(tree : list)
  tmp ← LIST(;)
  bin ← ↓;
  for feuille ∈ tree do
    if ISINSTANCE(feuille, list) then
      if bin ∈ feuille[0] then
        tmp = tmp + UNBIN(feuille[1 :]);
      else
        tmp.append(UNBIN(feuille))
    else
      tmp.append(feuille)
  return tmp

function LEXICALE2NAIRE(tree)
  if |tree| == 3 then
    head, leftchild, rightchild ← tree;
    if ↑ ∈ head then
      gauche, droite ← SPLIT(bin, head, 1)
      return [gauche, GENERALUNBIN([droite, leftchild, rightchild])]
    else
      return [head, LEXICALE2NAIRE(leftchild), LEXICALE2NAIRE(rightchild)]
  else if |tree| == 2 then
    head, child ← tree;
    if ↑ ∈ head then
      gauche, droite ← SPLIT(bin, head, 1)
      return [gauche, LEXICALE2NAIRE([droite, child])]
    else
      return [head, child]

function GENERALUNBIN(tree)
  firststep ← LEXICALE2NAIRE(tree)
  return GENERALUNBIN(firststep)
```

---