algorithm *count_crit_pairs*

```
inputs
  S, the current basis
  L, the current set of critical pairs of S
  f, a new polynomial to add to S
outputs
  the number m of new critical pairs of minimal degree d generated by f relative to S and L
do
  let t = lm(f)
  let m = d = 0
  — First count number of new pairs that survive the criteria
  for i = 1,...|S| do
    let u = lm(S_i)
    if t, u are not coprime then
      let v = lcm(t, u)
      — Terminating at i − 1 should solve the Buchberger triplet problem
      — Easy optimization: avoid a loop by rewriting the following two ifs
      if lm(S_j) ∤ v ∀j = 1,...,i − 1 then
        if d = 0 or deg v < d then
          let d = deg v
          let m = 1
        else if deg v = d then
          add 1 to m
  — Now count number of old pairs that survive the criteria
  for (p, q) ∈ L do
    let u = lcm(lm(p), lm(q))
    let e = deg(u)
    if d = 0 or e ≤ d then
      if lcm(t, lm(s)) ∤ u ∀s ∈ S then
        if d = 0 or e < d then
          let d = e
          let m = 1
        else if e = d then
          add 1 to m
  return m, d
```