Algorithm 1a

D_sums[] := 0 (sum of D-values for each possible score)
D_max := global maximum D-value
for each family of contingency tables with fixed 2x2 sums
  D_fam := maximum D-value in family
  u_opt := value of n_++ achieving D_fam
  if D_fam < eps * D_max
    continue
  else
    D := D_fam, u := u_opt
    while D >= eps * D_max
      S := score of table with n_++ = u
      update D and add to D_sums[S]
      u := u+1
    end
    D := D_fam, u := u_opt
    while D >= eps * D_max
      u := u-1
      S := score of table with n_++ = u
      update D and add to D_sums[S]
    end
  end
end

Algorithm 1b

D_sums[] := 0 (sum of D-values for each possible score)
D_max := global maximum D-value
for each family of contingency tables with fixed 3x2 sums
  D_fam := maximum D-value in family
  if D_fam < eps * D_max
    continue
  else
    for each contingency table in family
      D := D-value of table
      S := score of table
      D_sums[S] := D_sums[S] + D
    end
  end