Algorithm 1: *ComputeAlpha* (for single element in the batch)

Data:
$out_{|\hat{A}| \times T}$ (result of softmax), where $|\hat{A}|$ (alphabet size),
$T = \bar{W}_{unpadded}/4$;
label (encoded by alphabet);
$bl = 0$ (blank index)
begin
  $S = len(label) * 2 + 1$
  $T = out.shape[1]$
  $a = zeros(S, T)$
  $a[0][0] = out[bl][0]$
  $a[1][0] = out[label[0]][0]$
  $c = a[0][0] + a[1][0]$
  if $c > 0$ **then**
    $a[0][0] = \frac{a[0][0]}{c}$
    $a[1][0] = \frac{a[1][0]}{c}$
  for $t := 1$ **to** T **do**
    $start = \max(0, S - 2 \times (T - t))$
    $end = \min(2 \times t + 2, S)$
    for $s := start$ **to** end **do**
      $i = max(0, floor(\frac{s-1}{2}))$
      if $s \ mod \ 2 = 0$ **then**
        if $s = 0$ **then**
          $a[s][t] = a[s][t-1] \times out[bl][t]$
        else
          $a[s][t] = (a[s][t-1] + a[s-1][t-1]) \times out[bl][t]$
      else if $s = 1$ ***or*** $label[i] = label[i-1]$ **then**
        $a[s][t] = (a[s][t-1] + a[s-1][t-1]) \times out[label[i]][t]$
      else
        $a[s][t] =$
        $(a[s][t-1] + a[s-1][t-1] + a[s-2][t-1]) \times out[label[i]][t]$
    $c = \sum_{i=start}^{end} a[i][t]$
    if $c > 0$ **then**
      for $s := start$ **to** end **do**
        $a[s][t] = \frac{a[s][t]}{c}$
  return a