Algorithm 1 Computation of Trace Points from a given CIGAR-String

Input: $seq1, seq2, start\_seq1, end\_seq1, start\_seq2, \Delta, cigar$ mit
$|seq1|, |seq2|, |cigar| > 0$;
$start\_seq1, start\_seq2 \geq 0$;
$start\_seq1 < end\_seq1$ und
$\Delta > 0$

Output: Array TP of Trace Points

function *encode*($seq1, seq2, start\_seq1, end\_seq1, start\_seq2, \Delta, cigar$)
  $itv\_size \leftarrow MAX(1, \lceil start\_seq1/\Delta \rceil)$
  $itv\_count \leftarrow MIN(\lceil |seq1|/\Delta \rceil, \lceil |seq2|/\Delta \rceil)$
  for $i \leftarrow 0$ **upto** $|itv\_count|$ **do**

$$itv[i] \leftarrow \begin{cases} start\_seq1, itv\_size \cdot \Delta - 1 & \text{if } i = 0 \\ (itv\_size + i - 1) \cdot \Delta, (itv\_size + i) \cdot \Delta - 1 & \text{if } 0 < i < |itv\_count| \\ (itv\_size + i - 1) \cdot \Delta, end\_seq1 - 1 & \text{else.} \end{cases}$$

  end for
  $count1, count2, count3 \leftarrow 0$
  $TP \leftarrow$ Array for Trace Points
  for each (cig_count, cig_symbol) in cigar **do**
    for $i \leftarrow 0$ **upto** cig_count **do**
      if cig_symbol = 'I' **then**
        increment $count1$
      else if cig_symbol = 'D' **then**
        increment $count2$
      else
        increment $count1, count2$
      end if
      if count1 = intervals[count3][1] + 1 **and** count1 $\neq |seq1|$ **then**
        append (count2 - 1 + start_seq2) to TP
      end if
      if count $\neq |itv| - 1$ **then**
        increment $count3$
      end if
    end for
  end for
  return TP
end function

Algorithm 2 Computation of a CIGAR-String from a given Trace Point Array

Input: $seq1, seq2, \Delta, TP$ mit
$|seq1|, |seq2|, \Delta, |TP| > 0$
Output: CIGAR-String

function *decode*($seq1, seq2, \Delta, TP$)
    $cig \leftarrow$ empty String
    for $i \leftarrow 0$ **upto** $|TP|$ **do**
        append to cig:

$$\begin{cases} \textbf{cigar}(seq1[0...\Delta], seq2[0...TP[i]+1]) & \text{if } i = 0 \\ \textbf{cigar}(seq1[i \cdot \Delta ... |seq1|], seq2[TP[i-1]+1 ... |seq2|]) & \text{if } i = |TP| - 1 \\ \textbf{cigar}(seq1[i \cdot \Delta ... (i+1) \cdot \Delta], seq2[TP[i-1]+1] ... TP[i]+1) & \text{else.} \end{cases}$$

    end for
    $cig \leftarrow$ **combine**(cig)
    return cig
end function

function *combine*($cigar$)
    $cig \leftarrow$ empty String
    $tmp \leftarrow 0$
    for each (cig_count, cig_symbol) in cigar **do**
        $tmp \leftarrow tmp + previous\_cig\_count$
        if cig_symbol = previous_cig_symbol **then**
            if not last element in cigar **then**
                $tmp \leftarrow 0$
            end if
        end if
        if last element in cigar **then**
            append $(tmp + cig\_count, cig\_symbol)$ to cig
        end if
    end for
    return cig
end function