Pseudocode for the reconstruction of a flat zone

Waiting queue algorithm

```
/* x is the point (or pixel) of the input image I whose     */
/* flat zone is desired                                     */
/* I' is the output image, where the flat zone of x will    */
/* be labeled to LABEL_FZ                                   */
/*                                                          */
/* LABEL_NO_FZ must be different from LABEL_FZ, and, in     */
/* principle, LABEL_FZ should be different from 0.          */
/* For example, LABEL_FZ and LABEL_NO_FZ could be, 255, and */
/* 0, respectively.                                         */


PROCEDURE ReconstructFlatZone (I', I, x, LABEL_FZ)

  /* Output image I' is initialized */
  Set_image_value ( I', LABEL_NO_FZ)

  /* Value of pixel x in output image is labeled as flat zone */
  I' (x) = LABEL_FZ

  /* Creation of the flatzone waiting queue */
  flatzone = Createqueue()

  /* Initialization of flatzone with pixel x */
  Insert (flatzone, x)

  WHILE flatzone NOT EMPTY

    p = Extract (flatzone)

    FOR ALL neighbors p' of p DO


      IF ((I(p') == I(p)) AND (I'(p') != LABEL_FZ))
      THEN

        /* Pixel p' belongs to x's flat zone and has not been treated */

        /* Pixel p' is labeled as flat zone and inserted into the queue */
        I' (p') = LABEL_FZ
        Insert (flatzone, p')

      ENDIF

    ENDFOR

  ENDWHILE

ENDPROCEDURE
```

The function ReconstructFlatZone uses a standard waiting queue.
Let us mention that there exists another type of queue, the
'hierarchical' waiting queue in which queue elements belong
to several categories with different priorities.