This file presents the algorithm for a selection of a subset of low correlated bits from a matrix of bits, e.g. sketches. Algorithm is used in papers of Vladimir Mic, David Novak and Pavel Zezula about similarity search utilizing *sketches*.

We recommend to use parameters i and k set to 4,000 and 10 respectively. These params determine the time of searching.

Algorithm 1 Selection of low correlated bits

$Corr \leftarrow$ correlation matrix with absolute values
$p \leftarrow$ cardinality of returned subset
$minMaxCorr \leftarrow \infty$
$ret \leftarrow \emptyset$
Set values on diagonal $Corr(i, i)$ to zero
for `1 .. k` **do** ▷ k attempts
  $curIndexes \leftarrow$ random index
  for `1 .. i` **do** ▷ i iterations
    while $curIndexes.size < p$ **do** ▷ Try to create best possible answer
      set $bySum \leftarrow true$ with prob. 0.6
      $z \leftarrow$ index a with minimal value:
      if $bySum$ **then**
        $\sum_{b \in curIndexes} Corr(a, b)$
      else
        $\max_{b \in curIndexes} Corr(a, b)$
      $curIndexes \leftarrow curIndexes \cup \{z\}$
    $curMaxCorr \leftarrow \max_{a,b \in curIndexes} Corr(a, b)$
    if $curMaxCorr < minMaxCorr$ **then**
      $minMaxCorr \leftarrow curMaxCorr$
      $ret \leftarrow curIndexes$
    $r \leftarrow$ random number from $[0.02, 0.03]$
    while Exists $Corr(a, b) > curMaxCorr - r$ **do** ▷ remove bits with high correlations
      $aSum = \sum_{c \in curIndexes} Corr(a, c)$
      $bSum = \sum_{c \in curIndexes} Corr(b, c)$
      if $aSum > bSum$ **then**
        $curIndexes.remove(a)$
      else
        $curIndexes.remove(b)$
return ret