package com.twitter.search.core.earlybird.index.inverted;

/\*\*

\* A forward search finger used, optionally, by {@link SkipListContainer#search}.

\*

\* A search finger is pointer to the result returned by last time a search method is performed.

\* @see <a href="http://en.wikipedia.org/wiki/Finger\_search">Finger search wikipedia</a>.

\*

\* Using a search finger on a skip list could reduce the search search time from

\* log(n) to log(k), where n is length of the skip list and k is the distance between last searched

\* key and current searched key.

\*/

public class SkipListSearchFinger {

// Pointer used when initialize the search finger.

public static final int INITIAL\_POINTER = Integer.MIN\_VALUE;

private final int[] lastPointers;

/\*\*

\* Creates a new search finger.

\*/

public SkipListSearchFinger(int maxTowerHeight) {

lastPointers = new int[maxTowerHeight];

reset();

}

public void reset() {

for (int i = 0; i < lastPointers.length; i++) {

setPointer(i, INITIAL\_POINTER);

}

}

public int getPointer(int level) {

return lastPointers[level];

}

public void setPointer(int level, int pointer) {

lastPointers[level] = pointer;

}

public boolean isInitialPointer(int pointer) {

return pointer == INITIAL\_POINTER;

}

}