package com.twitter.search.common.encoding.features;

/\*\*

\* Encodes multiple values (bytes or bits) into an integer.

\*/

public class EncodedFeatures {

private int value;

public final void setSerializedValue(int val) {

this.value = val;

}

public final int getSerializedValue() {

return value;

}

// setByte is agnostic to signed / unsigned bytes.

protected final EncodedFeatures setByte(byte count, int bitshift, long inverseMask) {

value = (int) ((value & inverseMask) | ((count & 0xffL) << bitshift));

return this;

}

/\*\*

\* Sets the value but only if greater. setByteIfGreater assumes unsigned bytes.

\*/

public final EncodedFeatures setByteIfGreater(byte newCount, int bitshift, long inversemask) {

if ((getByte(bitshift) & 0xff) < (newCount & 0xff)) {

setByte(newCount, bitshift, inversemask);

}

return this;

}

protected final int getByte(int bitshift) {

return (int) (((value & 0xffffffffL) >>> bitshift) & 0xffL);

}

protected final int getByteMasked(int bitshift, long mask) {

return (int) (((value & mask) >>> bitshift) & 0xffL);

}

protected final EncodedFeatures setBit(int bit, boolean flag) {

if (flag) {

value |= bit;

} else {

value &= ~bit;

}

return this;

}

protected final boolean getBit(int bit) {

return (value & bit) != 0;

}

@Override

public String toString() {

return String.format("%x", value);

}

}