/\* ----------------------------------------------------------------------------

\* This file was automatically generated by SWIG (http://www.swig.org).

\* Version 4.0.2

\*

\* Do not make changes to this file unless you know what you are doing--modify

\* the SWIG interface file instead.

\* ----------------------------------------------------------------------------- \*/

package com.twitter.ann.faiss;

public class InvertedLists {

private transient long swigCPtr;

protected transient boolean swigCMemOwn;

protected InvertedLists(long cPtr, boolean cMemoryOwn) {

swigCMemOwn = cMemoryOwn;

swigCPtr = cPtr;

}

protected static long getCPtr(InvertedLists obj) {

return (obj == null) ? 0 : obj.swigCPtr;

}

@SuppressWarnings("deprecation")

protected void finalize() {

delete();

}

public synchronized void delete() {

if (swigCPtr != 0) {

if (swigCMemOwn) {

swigCMemOwn = false;

swigfaissJNI.delete\_InvertedLists(swigCPtr);

}

swigCPtr = 0;

}

}

public void setNlist(long value) {

swigfaissJNI.InvertedLists\_nlist\_set(swigCPtr, this, value);

}

public long getNlist() {

return swigfaissJNI.InvertedLists\_nlist\_get(swigCPtr, this);

}

public void setCode\_size(long value) {

swigfaissJNI.InvertedLists\_code\_size\_set(swigCPtr, this, value);

}

public long getCode\_size() {

return swigfaissJNI.InvertedLists\_code\_size\_get(swigCPtr, this);

}

public long list\_size(long list\_no) {

return swigfaissJNI.InvertedLists\_list\_size(swigCPtr, this, list\_no);

}

public SWIGTYPE\_p\_unsigned\_char get\_codes(long list\_no) {

long cPtr = swigfaissJNI.InvertedLists\_get\_codes(swigCPtr, this, list\_no);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_unsigned\_char(cPtr, false);

}

public LongVector get\_ids(long list\_no) {

return new LongVector(swigfaissJNI.InvertedLists\_get\_ids(swigCPtr, this, list\_no), false);

}

public void release\_codes(long list\_no, SWIGTYPE\_p\_unsigned\_char codes) {

swigfaissJNI.InvertedLists\_release\_codes(swigCPtr, this, list\_no, SWIGTYPE\_p\_unsigned\_char.getCPtr(codes));

}

public void release\_ids(long list\_no, LongVector ids) {

swigfaissJNI.InvertedLists\_release\_ids(swigCPtr, this, list\_no, SWIGTYPE\_p\_long\_long.getCPtr(ids.data()), ids);

}

public long get\_single\_id(long list\_no, long offset) {

return swigfaissJNI.InvertedLists\_get\_single\_id(swigCPtr, this, list\_no, offset);

}

public SWIGTYPE\_p\_unsigned\_char get\_single\_code(long list\_no, long offset) {

long cPtr = swigfaissJNI.InvertedLists\_get\_single\_code(swigCPtr, this, list\_no, offset);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_unsigned\_char(cPtr, false);

}

public void prefetch\_lists(LongVector list\_nos, int nlist) {

swigfaissJNI.InvertedLists\_prefetch\_lists(swigCPtr, this, SWIGTYPE\_p\_long\_long.getCPtr(list\_nos.data()), list\_nos, nlist);

}

public long add\_entry(long list\_no, long theid, SWIGTYPE\_p\_unsigned\_char code) {

return swigfaissJNI.InvertedLists\_add\_entry(swigCPtr, this, list\_no, theid, SWIGTYPE\_p\_unsigned\_char.getCPtr(code));

}

public long add\_entries(long list\_no, long n\_entry, LongVector ids, SWIGTYPE\_p\_unsigned\_char code) {

return swigfaissJNI.InvertedLists\_add\_entries(swigCPtr, this, list\_no, n\_entry, SWIGTYPE\_p\_long\_long.getCPtr(ids.data()), ids, SWIGTYPE\_p\_unsigned\_char.getCPtr(code));

}

public void update\_entry(long list\_no, long offset, long id, SWIGTYPE\_p\_unsigned\_char code) {

swigfaissJNI.InvertedLists\_update\_entry(swigCPtr, this, list\_no, offset, id, SWIGTYPE\_p\_unsigned\_char.getCPtr(code));

}

public void update\_entries(long list\_no, long offset, long n\_entry, LongVector ids, SWIGTYPE\_p\_unsigned\_char code) {

swigfaissJNI.InvertedLists\_update\_entries(swigCPtr, this, list\_no, offset, n\_entry, SWIGTYPE\_p\_long\_long.getCPtr(ids.data()), ids, SWIGTYPE\_p\_unsigned\_char.getCPtr(code));

}

public void resize(long list\_no, long new\_size) {

swigfaissJNI.InvertedLists\_resize(swigCPtr, this, list\_no, new\_size);

}

public void reset() {

swigfaissJNI.InvertedLists\_reset(swigCPtr, this);

}

public void merge\_from(InvertedLists oivf, long add\_id) {

swigfaissJNI.InvertedLists\_merge\_from(swigCPtr, this, InvertedLists.getCPtr(oivf), oivf, add\_id);

}

public double imbalance\_factor() {

return swigfaissJNI.InvertedLists\_imbalance\_factor(swigCPtr, this);

}

public void print\_stats() {

swigfaissJNI.InvertedLists\_print\_stats(swigCPtr, this);

}

public long compute\_ntotal() {

return swigfaissJNI.InvertedLists\_compute\_ntotal(swigCPtr, this);

}

static public class ScopedIds {

private transient long swigCPtr;

protected transient boolean swigCMemOwn;

protected ScopedIds(long cPtr, boolean cMemoryOwn) {

swigCMemOwn = cMemoryOwn;

swigCPtr = cPtr;

}

protected static long getCPtr(ScopedIds obj) {

return (obj == null) ? 0 : obj.swigCPtr;

}

@SuppressWarnings("deprecation")

protected void finalize() {

delete();

}

public synchronized void delete() {

if (swigCPtr != 0) {

if (swigCMemOwn) {

swigCMemOwn = false;

swigfaissJNI.delete\_InvertedLists\_ScopedIds(swigCPtr);

}

swigCPtr = 0;

}

}

public void setIl(InvertedLists value) {

swigfaissJNI.InvertedLists\_ScopedIds\_il\_set(swigCPtr, this, InvertedLists.getCPtr(value), value);

}

public InvertedLists getIl() {

long cPtr = swigfaissJNI.InvertedLists\_ScopedIds\_il\_get(swigCPtr, this);

return (cPtr == 0) ? null : new InvertedLists(cPtr, false);

}

public void setIds(LongVector value) {

swigfaissJNI.InvertedLists\_ScopedIds\_ids\_set(swigCPtr, this, SWIGTYPE\_p\_long\_long.getCPtr(value.data()), value);

}

public LongVector getIds() {

return new LongVector(swigfaissJNI.InvertedLists\_ScopedIds\_ids\_get(swigCPtr, this), false);

}

public void setList\_no(long value) {

swigfaissJNI.InvertedLists\_ScopedIds\_list\_no\_set(swigCPtr, this, value);

}

public long getList\_no() {

return swigfaissJNI.InvertedLists\_ScopedIds\_list\_no\_get(swigCPtr, this);

}

public ScopedIds(InvertedLists il, long list\_no) {

this(swigfaissJNI.new\_InvertedLists\_ScopedIds(InvertedLists.getCPtr(il), il, list\_no), true);

}

public LongVector get() {

return new LongVector(swigfaissJNI.InvertedLists\_ScopedIds\_get(swigCPtr, this), false);

}

}

static public class ScopedCodes {

private transient long swigCPtr;

protected transient boolean swigCMemOwn;

protected ScopedCodes(long cPtr, boolean cMemoryOwn) {

swigCMemOwn = cMemoryOwn;

swigCPtr = cPtr;

}

protected static long getCPtr(ScopedCodes obj) {

return (obj == null) ? 0 : obj.swigCPtr;

}

@SuppressWarnings("deprecation")

protected void finalize() {

delete();

}

public synchronized void delete() {

if (swigCPtr != 0) {

if (swigCMemOwn) {

swigCMemOwn = false;

swigfaissJNI.delete\_InvertedLists\_ScopedCodes(swigCPtr);

}

swigCPtr = 0;

}

}

public void setIl(InvertedLists value) {

swigfaissJNI.InvertedLists\_ScopedCodes\_il\_set(swigCPtr, this, InvertedLists.getCPtr(value), value);

}

public InvertedLists getIl() {

long cPtr = swigfaissJNI.InvertedLists\_ScopedCodes\_il\_get(swigCPtr, this);

return (cPtr == 0) ? null : new InvertedLists(cPtr, false);

}

public void setCodes(SWIGTYPE\_p\_unsigned\_char value) {

swigfaissJNI.InvertedLists\_ScopedCodes\_codes\_set(swigCPtr, this, SWIGTYPE\_p\_unsigned\_char.getCPtr(value));

}

public SWIGTYPE\_p\_unsigned\_char getCodes() {

long cPtr = swigfaissJNI.InvertedLists\_ScopedCodes\_codes\_get(swigCPtr, this);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_unsigned\_char(cPtr, false);

}

public void setList\_no(long value) {

swigfaissJNI.InvertedLists\_ScopedCodes\_list\_no\_set(swigCPtr, this, value);

}

public long getList\_no() {

return swigfaissJNI.InvertedLists\_ScopedCodes\_list\_no\_get(swigCPtr, this);

}

public ScopedCodes(InvertedLists il, long list\_no) {

this(swigfaissJNI.new\_InvertedLists\_ScopedCodes\_\_SWIG\_0(InvertedLists.getCPtr(il), il, list\_no), true);

}

public ScopedCodes(InvertedLists il, long list\_no, long offset) {

this(swigfaissJNI.new\_InvertedLists\_ScopedCodes\_\_SWIG\_1(InvertedLists.getCPtr(il), il, list\_no, offset), true);

}

public SWIGTYPE\_p\_unsigned\_char get() {

long cPtr = swigfaissJNI.InvertedLists\_ScopedCodes\_get(swigCPtr, this);

return (cPtr == 0) ? null : new SWIGTYPE\_p\_unsigned\_char(cPtr, false);

}

}

public final static long INVALID\_CODE\_SIZE = swigfaissJNI.InvertedLists\_INVALID\_CODE\_SIZE\_get();

}