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

\* 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 OperatingPoints {

private transient long swigCPtr;

protected transient boolean swigCMemOwn;

protected OperatingPoints(long cPtr, boolean cMemoryOwn) {

swigCMemOwn = cMemoryOwn;

swigCPtr = cPtr;

}

protected static long getCPtr(OperatingPoints 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\_OperatingPoints(swigCPtr);

}

swigCPtr = 0;

}

}

public void setAll\_pts(OperatingPointVector value) {

swigfaissJNI.OperatingPoints\_all\_pts\_set(swigCPtr, this, OperatingPointVector.getCPtr(value), value);

}

public OperatingPointVector getAll\_pts() {

long cPtr = swigfaissJNI.OperatingPoints\_all\_pts\_get(swigCPtr, this);

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

}

public void setOptimal\_pts(OperatingPointVector value) {

swigfaissJNI.OperatingPoints\_optimal\_pts\_set(swigCPtr, this, OperatingPointVector.getCPtr(value), value);

}

public OperatingPointVector getOptimal\_pts() {

long cPtr = swigfaissJNI.OperatingPoints\_optimal\_pts\_get(swigCPtr, this);

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

}

public OperatingPoints() {

this(swigfaissJNI.new\_OperatingPoints(), true);

}

public int merge\_with(OperatingPoints other, String prefix) {

return swigfaissJNI.OperatingPoints\_merge\_with\_\_SWIG\_0(swigCPtr, this, OperatingPoints.getCPtr(other), other, prefix);

}

public int merge\_with(OperatingPoints other) {

return swigfaissJNI.OperatingPoints\_merge\_with\_\_SWIG\_1(swigCPtr, this, OperatingPoints.getCPtr(other), other);

}

public void clear() {

swigfaissJNI.OperatingPoints\_clear(swigCPtr, this);

}

public boolean add(double perf, double t, String key, long cno) {

return swigfaissJNI.OperatingPoints\_add\_\_SWIG\_0(swigCPtr, this, perf, t, key, cno);

}

public boolean add(double perf, double t, String key) {

return swigfaissJNI.OperatingPoints\_add\_\_SWIG\_1(swigCPtr, this, perf, t, key);

}

public double t\_for\_perf(double perf) {

return swigfaissJNI.OperatingPoints\_t\_for\_perf(swigCPtr, this, perf);

}

public void display(boolean only\_optimal) {

swigfaissJNI.OperatingPoints\_display\_\_SWIG\_0(swigCPtr, this, only\_optimal);

}

public void display() {

swigfaissJNI.OperatingPoints\_display\_\_SWIG\_1(swigCPtr, this);

}

public void all\_to\_gnuplot(String fname) {

swigfaissJNI.OperatingPoints\_all\_to\_gnuplot(swigCPtr, this, fname);

}

public void optimal\_to\_gnuplot(String fname) {

swigfaissJNI.OperatingPoints\_optimal\_to\_gnuplot(swigCPtr, this, fname);

}

}