%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Test cases for Matlab Functions

% Section : DATA IMPORT AND EXPORT

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% 6. Scientific

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

dateobj = cdfepoch(now);

info = cdfinfo('example.cdf');

data = cdfread('example.cdf', 'Variable', {'Time'});

cdfwrite('example', {'Longitude', 0:360});

dstr2 = datestr(todatenum(obj)); %2 counts?

n = todatenum(obj);

names = cdflib.getConstantNames();

value = cdflib.getConstantValue(names{1});

mode = cdflib.getFileBackward();

copyright = cdflib.getLibraryCopyright();

[version, release, increment] = cdflib.getLibraryVersion();

mode = cdflib.getValidate();

cdflib.setFileBackward('BACKWARDFILEon');

cdflib.setValidate('VALIDATEFILEon');

cdflib.close(cdfid);

cdfId = cdflib.create('myfile.cdf');

cdflib.delete(cdfId);

numBuf = cdflib.getCacheSize(cdfid);

checksummode = cdflib.getChecksum(cdfid);

[ctype, cparms, cpercentage] = cdflib.getCompression(cdfId);

numBuf = cdflib.getCompressionCacheSize(cdfId);

copyright = cdflib.getCopyright(cdfId);

format = cdflib.getFormat(cdfId);

majority = cdflib.getMajority(cdfId);

name = cdflib.getName(cdfId);

mode = cdflib.getReadOnlyMode(cdfId);

numBuf = cdflib.getStageCacheSize(cdfId);

%numRecs = cdflib.getVarNumRecsWritten(cdfid,0);

%maxRecNum = cdflib.getVarsMaxWrittenRecNum(cdfid);

[version, release, increment] = cdflib.getVersion(cdfId);

info = cdflib.inquire(cdfId);

cdfId = cdflib.open('example.cdf');

numBuf = cdflib.getCacheSize(cdfid);

cdflib.setChecksum(cdfid,'MD5\_CHECKSUM');

cdflib.setCompression(cdfId,'HUFF\_COMPRESSION');

cdflib.setCompressionCacheSize(cdfId,100);

cdflib.setFormat(cdfId, 'MULTI\_FILE');

cdflib.setMajority(cdfId,'COLUMN\_MAJOR');

cdflib.setReadOnlyMode(cdfId,'READONLYon');

cdflib.setStageCacheSize(cdfId, 200);

cdflib.closeVar(cdfid, varnum);

varNum = cdflib.createVar(cdfid,'Time','cdf\_int1',1,[],true,[]);

cdflib.deleteVar(cdfid,0);

cdflib.deleteVarRecords(cdfid,varnum,1,2);

numrecs = cdflib.getVarAllocRecords(cdfid,0);

blockingFactor = cdflib.getVarBlockingFactor(cdfId,varNum);

numBuf = cdflib.getVarCacheSize(cdfid,varNum);

[ctype params percent] = cdflib.getVarCompression(cdfid,0);

datum = cdflib.getVarData(cdfId,3,0,[0 0 0]);

maxRecNum = cdflib.getVarMaxAllocRecNum(cdfid,0);

maxrec = cdflib.getVarMaxwrittenRecNum(cdfId,varNum);

maxRecNum = cdflib.getVarsMaxWrittenRecNum(cdfid);

name = cdflib.getVarName(cdfid,1);

varNum = cdflib.getVarNum(cdfid,'Longitude');

numRecs = cdflib.getVarNumRecsWritten(cdfid,0);

padval = cdflib.getVarPadValue(cdfid,0);

data = cdflib.getVarRecordData(cdfId,varNum,recNum);

percent = cdflib.getVarReservePercent(cdfId,varNum);

stype = cdflib.getVarSparseRecords(cdfId,varNum);

data = cdflib.hyperGetVarData(cdfid,0,[0 maxRecNum 1]);

cdflib.hyperPutVarData(cdfid,varNum,0,[],int8(98));

info = cdflib.inquireVar(cdfid,1);

datum = cdflib.getVarData(cdfid,varNum,0);

cdflib.putVarRecordData(cdfid,varNum,0,int8(98));

cdflib.renameVar(cdfid,varNum,'NewName');

cdflib.setVarAllocBlockRecords(cdfid,varNum,1,10);

cdflib.setVarBlockingFactor(cdfid,varNum,10);

cdflib.setVarCacheSize(cdfid,varNum,5);

cdflib.setVarCompression(cdfid,0,'GZIP\_COMPRESSION',8);

cdflib.setVarInitialRecs(cdfid,varNum,100);

cdflib.setVarPadValue(cdfid,varNum,int8(1));

cdflib.setVarReservePercent(cdfid,varNum, 80);

cdflib.setVarsCacheSize(cdfid,6);

cdflib.setVarSparseRecords(cdfid,varNum,'PAD\_SPARSERECORDS');

attrNum = cdflib.createAttr(cdfId,'Purpose','global\_scope');

cdflib.deleteAttr(cdfid,attrNum);

cdflib.deleteAttrEntry(cdfid,attrNum,0);

cdflib.deleteAttrgEntry(cdfid,attrNum,0);

entrynum = cdflib.getAttrMaxEntry(cdfid,3);

entrynum = cdflib.getAttrMaxgEntry(cdfid,0);

attrName = cdflib.getAttrName(cdfId,0);

attrNum = cdflib.getAttrNum(cdfId,attrName);

attrScope = cdflib.getAttrScope(cdfid,0);

nentries = cdflib.getNumAttrEntries(cdfId,attrNum);

nentries = cdflib.getNumAttrgEntries(cdfId,attrNum);

numAttrs = cdflib.getNumAttributes(cdfid);

ngatts = cdflib.getNumgAttributes(cdfId);

%info = cdflib.inquireAttr(cdfid,0);

%attrscope = cdflib.getAttrScope(cdfid,3);

%[dtype numel] = cdflib.inquireAttrgEntry(cdfid,0,0);

cdflib.putAttrEntry(cdfid,attrNum,0,'CDF\_CHAR','My Variable Attribute Test');

cdflib.putAttrgEntry(cdfid,attrNum,0,'CDF\_CHAR','My Test');

cdflib.renameAttr(cdfId,attrNum,newName);

epoch = cdflib.computeEpoch(timeval);

epoch16 = cdflib.computeEpoch16(timeval);

timevec = cdflib.epoch16Breakdown(epoch16);

timevec = cdflib.epochBreakdown(epoch);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%