﻿using System;

using Microsoft.VisualStudio.TestTools.UnitTesting;

using SecurityLibrary;

namespace SecurityPackageTest

{

[TestClass]

public class VignereTest

{

string mainPlain = "wearediscoveredsaveyourself";

string mainCipherRep = "zicvtwqngrzgvtwavzhcqyglmgj".ToUpper();

string mainCipherAuto = "zicvtwqngkzeiigasxstslvvwla".ToUpper();

string mainKey = "deceptive";

string newPlain = "MICHIGANTECHNOLOGICALUNIVERSITY".ToLower();

string newCipherRep = "TWWNPZOAASWNUHZBNWWGSNBVCSLYPMM".ToUpper();

string newCipherAuto = "TWWNPZOAFMEOVULBZMEHYIYWBMTSTNL".ToUpper();

string newKey = "HOUGHTON".ToLower();

[TestMethod]

public void RepVignereTestEnc1()

{

RepeatingkeyVigenere algorithm = new RepeatingkeyVigenere();

string cipher = algorithm.Encrypt(mainPlain, mainKey);

Assert.IsTrue(cipher.Equals(mainCipherRep, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void RepVignereTestDec1()

{

RepeatingkeyVigenere algorithm = new RepeatingkeyVigenere();

string plain = algorithm.Decrypt(mainCipherRep, mainKey);

Assert.IsTrue(plain.Equals(mainPlain, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void RepVignereTestAnalysis1()

{

RepeatingkeyVigenere algorithm = new RepeatingkeyVigenere();

string key = algorithm.Analyse(mainPlain, mainCipherRep);

Assert.IsTrue(key.Equals(mainKey, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void AutoVignereTestEnc1()

{

AutokeyVigenere algorithm = new AutokeyVigenere();

string cipher = algorithm.Encrypt(mainPlain, mainKey);

Assert.IsTrue(cipher.Equals(mainCipherAuto, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void AutoVignereTestDec1()

{

AutokeyVigenere algorithm = new AutokeyVigenere();

string plain = algorithm.Decrypt(mainCipherAuto, mainKey);

Assert.IsTrue(plain.Equals(mainPlain, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void AutoVignereTestAnalysis1()

{

AutokeyVigenere algorithm = new AutokeyVigenere();

string key = algorithm.Analyse(mainPlain, mainCipherAuto);

Assert.IsTrue(key.Equals(mainKey, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void RepVignereTestNewEnc()

{

RepeatingkeyVigenere algorithm = new RepeatingkeyVigenere();

string cipher = algorithm.Encrypt(newPlain, newKey);

Assert.IsTrue(cipher.Equals(newCipherRep, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void RepVignereTestNewDec()

{

RepeatingkeyVigenere algorithm = new RepeatingkeyVigenere();

string plain = algorithm.Decrypt(newCipherRep, newKey);

Assert.IsTrue(plain.Equals(newPlain, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void RepVignereTestNewAnalysis()

{

RepeatingkeyVigenere algorithm = new RepeatingkeyVigenere();

string key = algorithm.Analyse(newPlain, newCipherRep);

Assert.IsTrue(key.Equals(newKey, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void AutoVignereTestNewEnc()

{

AutokeyVigenere algorithm = new AutokeyVigenere();

string cipher = algorithm.Encrypt(newPlain, newKey);

Assert.IsTrue(cipher.Equals(newCipherAuto, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void AutoVignereTestNewDec()

{

AutokeyVigenere algorithm = new AutokeyVigenere();

string plain = algorithm.Decrypt(newCipherAuto, newKey);

Assert.IsTrue(plain.Equals(newPlain, StringComparison.InvariantCultureIgnoreCase));

}

[TestMethod]

public void AutoVignereTestNewAnalysis()

{

AutokeyVigenere algorithm = new AutokeyVigenere();

string key = algorithm.Analyse(newPlain, newCipherAuto);

Assert.IsTrue(key.Equals(newKey, StringComparison.InvariantCultureIgnoreCase));

}

}

}