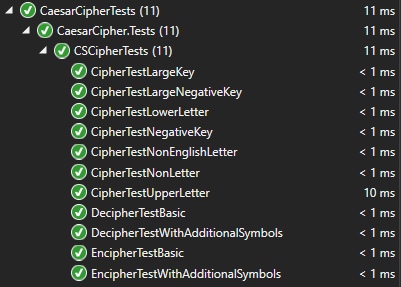
**Caesar Cipher tests**

**Test results:**



**Test methods:**

[TestMethod()]

public void CipherTestUpperLetter()

{

char upperLetter = 'A';

int key = 1;

char rezult = CSCipher.Cipher(upperLetter, key);

Assert.AreEqual('B', rezult);

}

[TestMethod()]

public void CipherTestLowerLetter()

{

char upperLetter = 'a';

int key = 1;

char rezult = CSCipher.Cipher(upperLetter, key);

Assert.AreEqual('b', rezult);

}

[TestMethod()]

public void CipherTestNonLetter()

{

char upperLetter = '5';

int key = 1;

char rezult = CSCipher.Cipher(upperLetter, key);

Assert.AreEqual('5', rezult);

}

[TestMethod()]

public void CipherTestLargeKey()

{

char upperLetter = 'A';

int key = 53;

char rezult = CSCipher.Cipher(upperLetter, key);

Assert.AreEqual('B', rezult);

}

[TestMethod()]

public void CipherTestNegativeKey()

{

char upperLetter = 'A';

int key = -1;

char rezult = CSCipher.Cipher(upperLetter, key);

Assert.AreEqual('Z', rezult);

}

[TestMethod()]

public void CipherTestLargeNegativeKey()

{

char upperLetter = 'A';

int key = -1015;

char rezult = CSCipher.Cipher(upperLetter, key);

Assert.AreEqual('Z', rezult);

}

[TestMethod()]

public void CipherTestNonEnglishLetter()

{

char upperLetter = 'Ą';

int key = 1;

char rezult = CSCipher.Cipher(upperLetter, key);

Assert.AreEqual('Ą', rezult);

}

[TestMethod()]

public void EncipherTestBasic()

{

string message = "abc";

int key = 3;

string rezult = CSCipher.Encipher(message, key);

Assert.AreEqual("def", rezult);

}

[TestMethod()]

public void EncipherTestWithAdditionalSymbols()

{

string message = "a b;c1";

int key = 3;

string rezult = CSCipher.Encipher(message, key);

Assert.AreEqual("d e;f1", rezult);

}

[TestMethod()]

public void DecipherTestBasic()

{

string message = "def";

int key = 3;

string rezult = CSCipher.Decipher(message, key);

Assert.AreEqual("abc", rezult);

}

[TestMethod()]

public void DecipherTestWithAdditionalSymbols()

{

string message = "d e;f1";

int key = 3;

string rezult = CSCipher.Decipher(message, key);

Assert.AreEqual("a b;c1", rezult);

}