San José State University
Department of Computer Science

Ahmad Yazdankhah

ahmad.yazdankhah@sjsu.edu
www.cs.sjsu.edu/~yazdankhah

CryptoUtil User Manual

Version 1.4

CS 265

Cryptography and Computer Security

Spring 2020

Objective

This document describes briefly the contents of CryptoUtil and it can be used. Along with this doc, a JavaDoc is provided that contains full detail of the classes, methods, and constants.

Abbreviations

Abbreviation	Stands For
Abs	Abstract
Arr	Array
Bin	Binary
Char	Character
Int	Interface
Perm	Permutation
Str	String
Sub	Substitution
Sys	System
Trans	Transposition
Util	Utility

General Rules

1. Mutable Methods

All mutable methods are suffixed with M.

Example 1

```
Word w0 = Word.constructFromHexStr("01234567");

Word w1 = Word.constructFromHexStr("89abcdef");

w0.xorM(w1);

w0.printHexStr("w0 after xorM");
```

Output: `[88888888] w0 after xorM`

This code XORs w0 and w1 and **puts the result in w0**. Therefore, w0 has been affected as its print shows.

Example 2

```
Word w0 = Word.constructFromHexStr("01234567");
Word w1 = Word.constructFromHexStr("89abcdef");
w0.xor(w1);
w0.printHexStr("w0 after xor");
```

Output:

```
[01234567] w0 after xor
```

This code XORs w0 and w1 and **puts the result in a new Word object**. Therefore, w0 value is not changed and is still pointing to its first value "01234567".

This version of XOR is useful if you want to put the result of an operation into another variable and you don't want to change the original variable.

Example 3

```
Word w0 = Word.constructFromHexStr("01234567");
Word w1 = Word.constructFromHexStr("89abcdef");
Word w2 = w0.xor(w1);
w0.printHexStr("w0 after xor");
w2.printHexStr("w2");
```

Output:

```
[01234567] w0 after xor
[88888888] w2
```

2. Mutable Method Arguments

In general, we'd like our methods not change the contents of the object we passed to them. But in some cases, this is inevitable and the contents of the passed objects changed. To distinguish when a method changes the passed object, we suffix the parameter name with "M".

Example 4

```
public void CRC(Word vM, UByte b) { ... }
```

This method will change the contents of "vM" parameter but does not change the contents of "b".

3. No Checking

Generally, there is no checking for the given parameters of methods and this is the callers of the methods' responsibility to satisfy the "**preconditions**" of the methods.

Use the provided JavaDoc for more information about the preconditions.

Datatypes

Primitive Datatypes

Datatype	**Bits**	**Comment**	**Underlying Data Structure**
Bit	1		boolean
Nibble	4		Bit[ ]
UByte	8	unsigned byte	Bit[ ]
DByte	16	double byte	UByte[ ]
Word	32		UByte[ ]
DWord	64	double word	Word[ ]
Quad	128	4 words	Word[ ]

Matrix Datatypes

Datatype	**Comment**	**Underlying Data Structure**
CharMatrix		char[] []
UByteMatrix		UByte[] []
LookupTable	A special UByteMatrix for some algorithms	UByteMatrix

Shift Registers Datatypes

Datatype	Comment	Underlying Data Structure
BSR	bit shift register	Bit[ ]
LFSR	linear feedback shift register	BSR
CSR	character shift register	char[ ]

Utilities

Cipher Utilities

Utility	Comment
ConversionUtil	converts some data structures to other data structures
StringUtil	operations on strings
FileUtil	operations on files
PrintUtil	facilities for printing strings, arrays and more
AnalyzerUtil	facilities for analyzing the content of a file or a string
Function	functions used in some algorithms
GeneralUtil	misc. utilities