

TAWHIDIC PARADIGM IN TEACHING & LEARNING: REINTEGRATING FAITH AND KNOWLEDGE

```
16:37:03) [MSC v.1929 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more info
```

```
IPython 8.30.0 -- An enhanced Interactive Python
```

```
In [1]: plaintext = "cryptography"
```

```
In [2]: plist = list(plaintext)
```

```
In [3]: pint = [ord(ch)-ord('a') for ch in plist]
```

```
In [4]: import numpy as np
```

```
In [5]: P = np.array(pint)
```

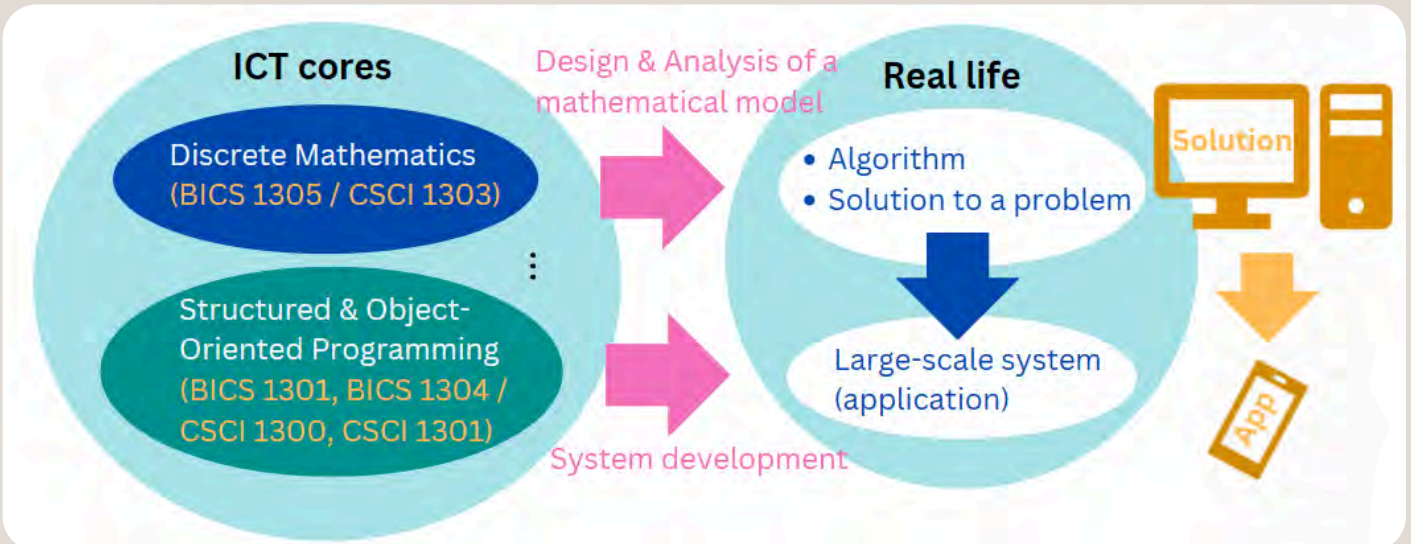
```
In [6]:
```



DR. TAKUMI SASE
DEPARTMENT OF COMPUTER
SCIENCE
TAKUMI@IIUM.EDU.MY

Learning Cryptography

Integration of Mathematics and Programming



Introduction

Cryptography is the science of secret writing. This is a field integrating mathematics and computer science technologies to secure our life. Hence, learning this field is a good opportunity for CS students to see how foundational ICT knowledge (such as discrete mathematics & programming) can be applied to explore a broader and more impactful area that addresses real-life problems. Learning Cryptography will tell us how a system we want to create can be modeled mathematically, how the modeled system is implemented on a computer through programming, and how it can appear in real-life use.

```
Output - CryptoClassical (run) x
run:
Enter a plaintext: Hello there.
Enter a key: 3
Encryption: Khoor wk huh.
Decryption: Hello there.
BUILD SUCCESSFUL (total time: 10 seconds)
```

```
Console 1/A x
Python 3.13.5 | packaged by Anaconda, Inc. | (main, Jun 12 2025, 16:37:03) [MSC v.1929 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.30.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: plaintext = "cryptography"
In [2]: plist = list(plaintext)
In [3]: pint = [ord(ch)-ord('a') for ch in plist]
In [4]: import numpy as np
In [5]: P = np.array(pint)
In [6]: K = np.array([[1,2],[3,4]])
In [7]:
```

Name	Type	Size	Value
K	Array of int64	[2, 2]	[[1 2] [3 4]]
P	Array of int64	[12]	[2 17 24 ... 15 ...]
pint	list	12	[2, 17, 24, 15, 19...]
plaintext	str	12	cryptography
plist	list	12	['c', 'r', 'y', 'p...]

Java Programming

KICT students will learn Java in the 1st semester of their 1st year. Therefore, learning Cryptography with Java can be a good start. For example, Caesar cipher is defined as

$$y = (x + k) \bmod 26$$

where x is a plaintext letter (0 to 25) and k is a key (any integer), and transforming this cipher into a representation on Java source code would be an important process. We can see how abstraction in mathematics (such as variables and functions) is helpful in implementation and this idea can help understand AES.

Python

Python enables us to easily interact with a computer. For example, coding Hill cipher with Python can be a good practice. Given a situation where plaintext is very long and the key matrix is of large size, NumPy library can show high computational power.

Future work

Cryptography is a large area. Important topics need to be selected from classical encryption techniques until recent cryptographic protocols. Quantum Cryptography will also be necessary to be introduced for the future.

