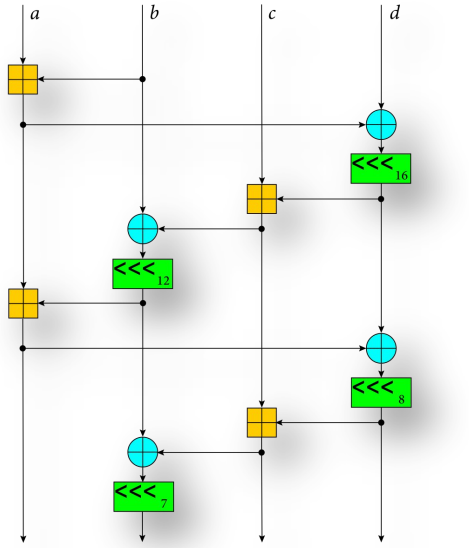
ChaCha20

Cryptography

"expa"	"nd 3"	"2–by"	"te k"
Key	Key	Key	Key
Key	Key	Key	Key
Pos.	Pos.	Nonce	Nonce



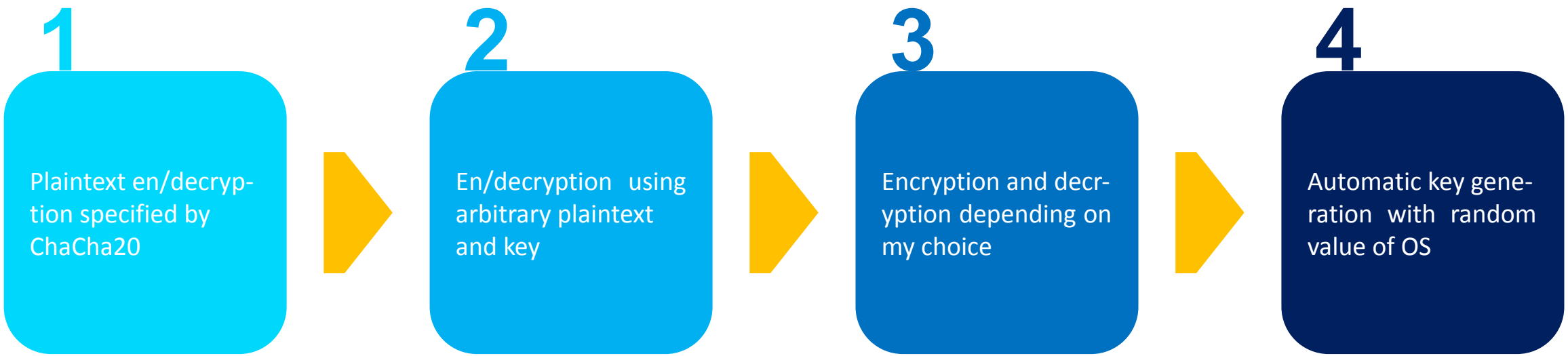
Yang, Heetak (Leslie)

Part 1,

PROGRESS

What is the algorithm used in this case study, and how does it work?

BASIC GOAL

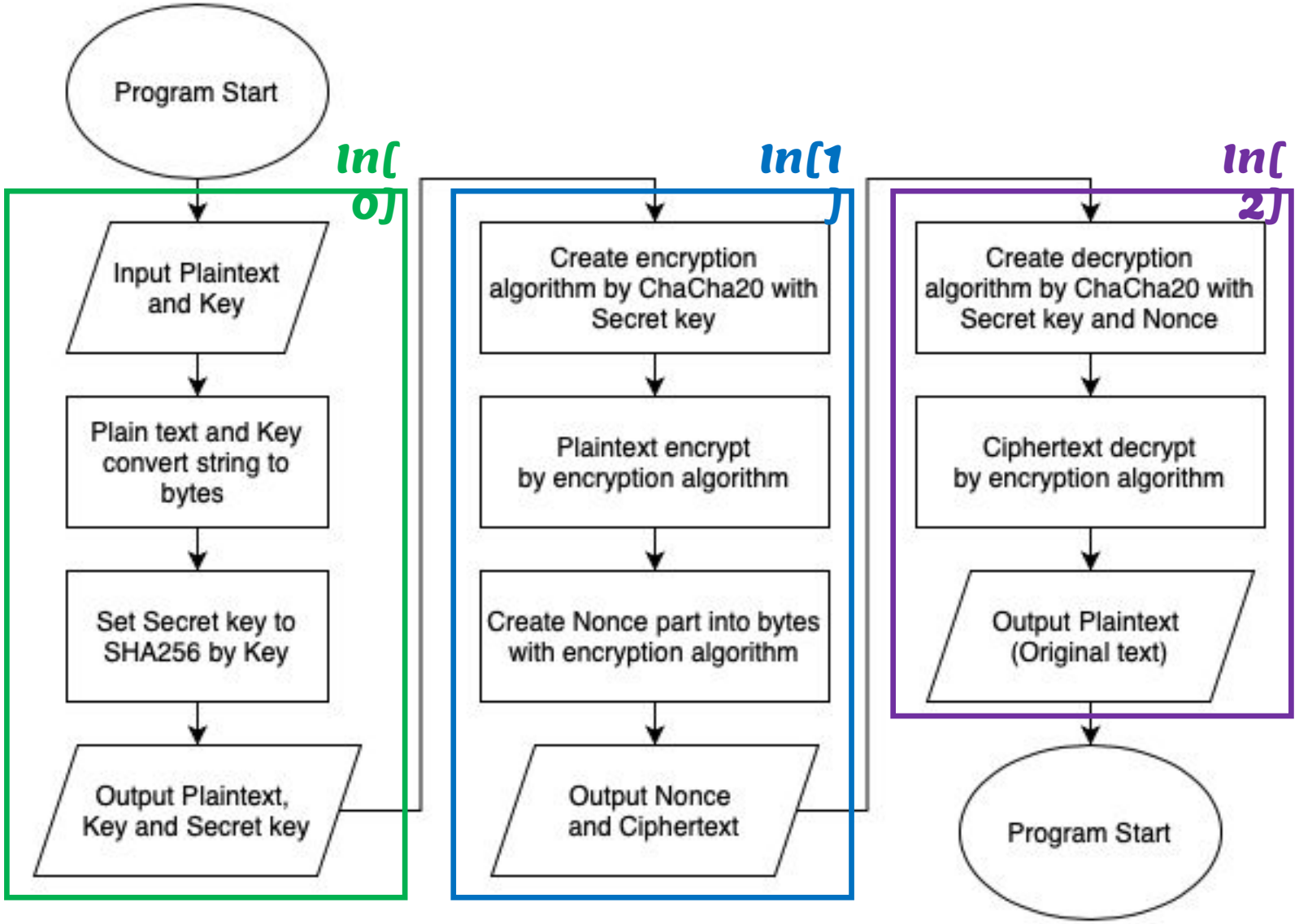


Part 2,

PROGRESS

What is the algorithm used in this case study, and how does it work?

FLOWCHART



Part 3,

CODING

What is the algorithm used in this case study, and how does it work?

DEPENDENCY

Name	Detail	Version
Python 3	Programming Language	3.9.2
hashlib	Hash Algorithms	python3
pycrytodome	Cryptographic library for Python	3.10.1

On Windows, you may need “Visual Studio C++ Builder”

- **Crypto.Cipher**
  - ChaCha20
  - Poly1305

Part 4,

CODING

What is the algorithm used in this case study, and how does it work?

CODE

```
# In[1]
# Indexing part
import hashlib, os
from base64 import b64encode
from Crypto.Cipher import ChaCha20


# Enter plaintext and key
print("\n======Cipher Program: ChaCha20======\n")
plaintextstr = input("Enter plaintext: ")
keystr = input("Enter secretkey: ")

# Convert value to bytes
plaintext = plaintextstr.encode()
key = keystr.encode()
secretkey = hashlib.sha256()
secretkey.update(key)
keyused = str(b64encode(secretkey.digest()),'utf-8’)

# Check working
print("\nPlain key:\t",plaintextstr)
print("Secret key:\t",keystr)
print("Key used:\t",keyused)
```

```
# In[1]
# Encrypt part
cipheralg = ChaCha20.new(key=secretkey.digest())
ciphertext = cipheralg.encrypt(plaintext)
nonce = str(b64encode(cipheralg.nonce),'utf-8')
cipheredtext = str(b64encode(ciphertext),'utf-8')
print("\n\n---------Encrypt(ChaCha20)------------")
print("Nonce code:\t",nonce)
print("Ciphertext:\t",cipheredtext)


# In[2]
# Decrypt part
cipheralg =
ChaCha20.new(key=secretkey.digest(),nonce=cipheralg.nonce)
plaintext = cipheralg.decrypt(ciphertext)
plaintextstr = str(plaintext,'utf-8')
print("\n\n---------Decrypt(ChaCha20)------------")
print("Decrypted:\t",plaintextstr)


# In[3]
# End part
print("\n\n\nFinish!")
```

Part 4,

CODING

What is the algorithm used in this case study, and how does it work?

CODE

```
import hashlib, os
from base64 import b64encode
from Crypto.Cipher import ChaCha20

# Enter plaintext and key
print("\n======Cipher Program: ChaCha20======\n")
plaintextstr = input("Enter plaintext: ")
keystr = input("Enter secretkey: ")

# Convert value to bytes
plaintext = plaintextstr.encode()
key = keystr.encode()
secretkey = hashlib.sha256()
secretkey.update(key)
keyused = str(b64encode(secretkey.digest()),'utf-8')
#keyused = b64encode(secretkey.digest()).decode('utf-8')

# Check working
print("\nPlain key:\t",plaintextstr)
print("Secret key:\t",keystr)
print("Key used:\t",keyused)
```

Part 4,

CODING

What is the algorithm used in this case study, and how does it work?

CODE

```
cipheralg = ChaCha20.new(key=secretkey.digest())
ciphertext = cipheralg.encrypt(plaintext)
nonce = str(b64encode(cipheralg.nonce),'utf-8')
cipheredtext = str(b64encode(ciphertext),'utf-8')

print("\n\n---------Encrypt(ChaCha20)------------")
print("Nonce code:\t",nonce)
print("Ciphertext:\t",cipheredtext)
```

```
cipheralg = ChaCha20.new(key=secretkey.digest(), nonce=cipheralg.nonce)
plaintext = cipheralg.decrypt(ciphertext)
plaintextstr = str(plaintext,'utf-8')

print("\n\n---------Decrypt(ChaCha20)------------")
print("Decrypted:\t",plaintextstr)
```

Part 4,

CODING

What is the algorithm used in this case study, and how does it work?

RESULT

```
======Cipher Program: ChaCha20======

Enter plaintext: I love Cybersecurity :)
Enter secretkey: Cryptography

Plain key:        I love Cybersecurity :)
Secret key:       Cryptography
Key used:         tYTuxyhUis7VpmwCZ91SCgCHG157c1stggL4Zxn2GFc=

---------Encrypt(ChaCha20)------------
Nonce code:       WH0cvpP/mvU=
Ciphertext:       ZyLgZmDmeIGSyO+ZE0cspincsZIxA5U=

---------Decrypt(ChaCha20)------------
Decrypted:        I love Cybersecurity :)


Finish!
Press Enter...
```

“ *THANK YOU* ”

- THE END -

CONCLUSION

- **Use ChaCha20 with SHA256**
- **Complete basic code**
- **Not finish to encryption and decryption depending on my choice**
- **Not yet to auto key generate**
- **It plans to complete the test of sending and receiving through communication by TLS, if Basic goal finish**

References

✔ *https://pycryptodome.readthedocs.io*

✔ *https://pypi.org/project/chacha20poly1305*