BASIC PYTHON

a logic gate in Python program

MAHASARAKHAM
UNIVERSITY

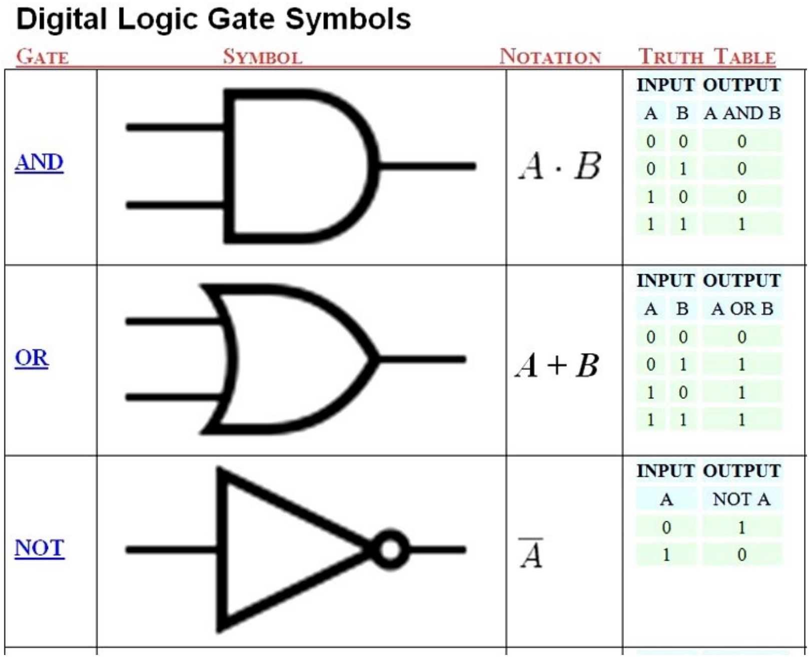
พหูนํ ปณฺ ฑิโต ชีเว

BASIC PYTHON

a logic gate in Python program

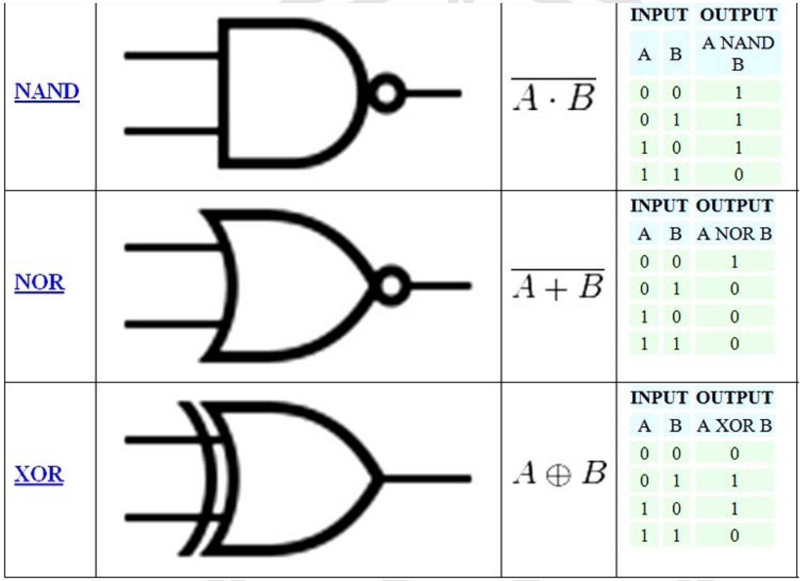
Olarik Surinta, PhD
Lecturer

Logic gates



Digital Logic Gate Symbols

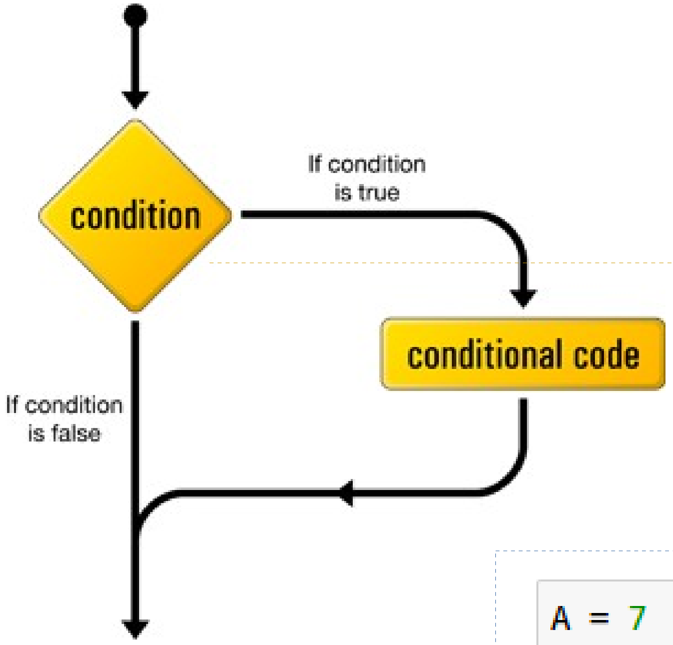
Gate	Symbol	Notation	Truth Table
AND		$A \cdot B$	INPUT OUTPUT A B A AND B 0 0 0 0 1 0 1 0 0 1 1 1
OR		$A + B$	INPUT OUTPUT A B A OR B 0 0 0 0 1 1 1 0 1 1 1 1
NOT		$\overline{A}$	INPUT OUTPUT A NOT A 0 1 1 0

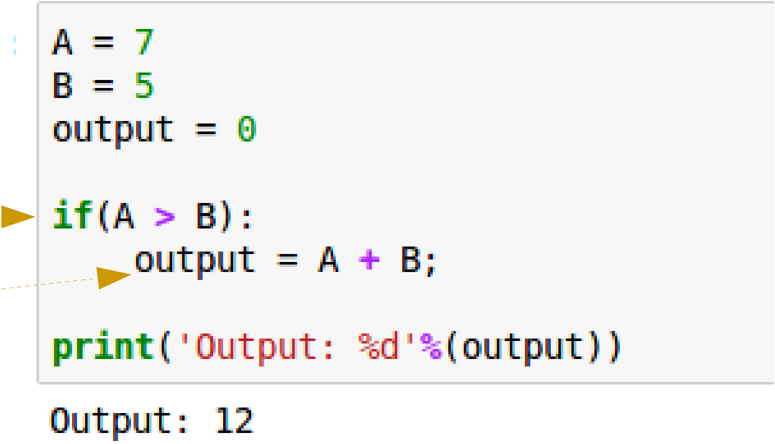


NAND		$\overline{A \cdot B}$	INPUT OUTPUT A B A NAND B 0 0 1 0 1 1 1 0 1 1 1 0
NOR		$\overline{A + B}$	INPUT OUTPUT A B A NOR B 0 0 1 0 1 0 1 0 0 1 1 0
XOR		$A \oplus B$	INPUT OUTPUT A B A XOR B 0 0 0 0 1 1 1 0 1 1 1 0

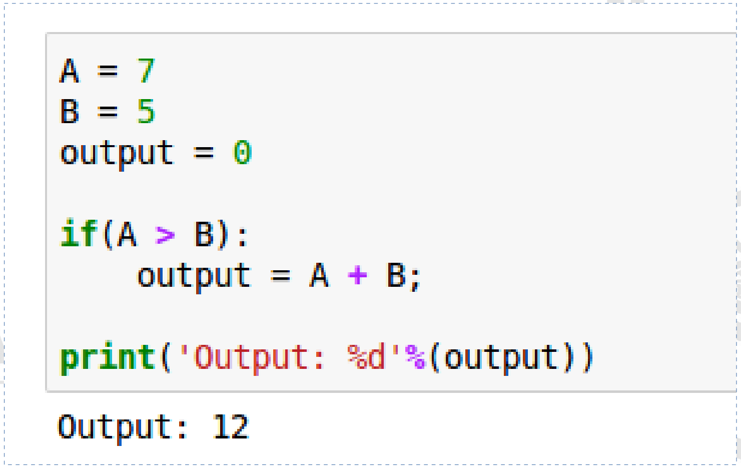
Cr. https://forums.autodesk.com/t5/autocad-electrical-forum/autocad-electrical-logic-circuit-help/td-p/6706653

if statement




```
A = 7
B = 5
output = 0

if(A > B):
    output = A + B;

print('Output: %d'%(output))
```

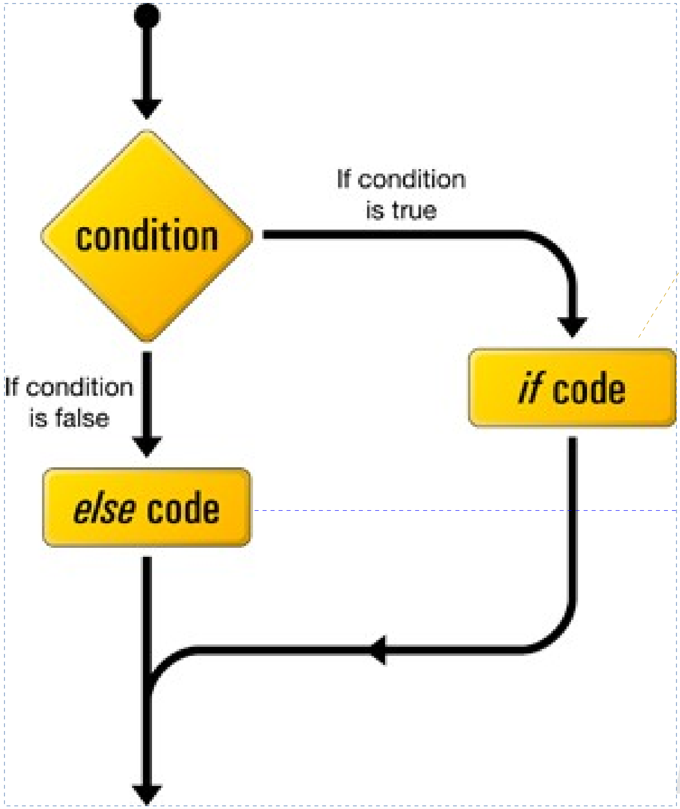
```
Output: 12
```

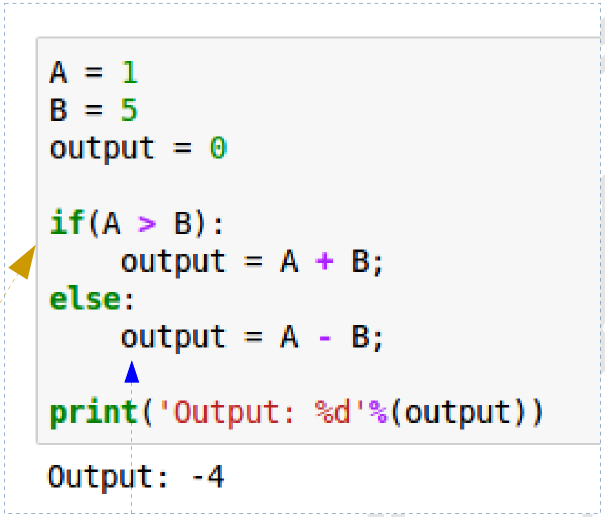


```
A = 7
B = 5
output = 0

if(A > B):
    output = A + B;

print('Output: %d'%(output))
```

```
Output: 12
```

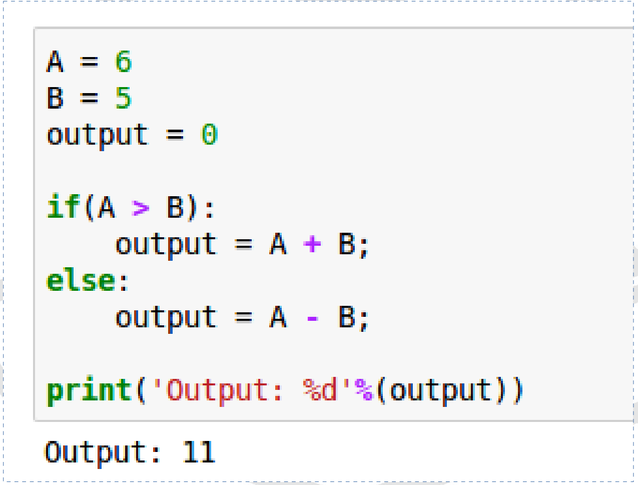
if..else statement





```
A = 1
B = 5
output = 0

if(A > B):
    output = A + B;
else:
    output = A - B;

print('Output: %d'%(output))
```

```
Output: -4
```



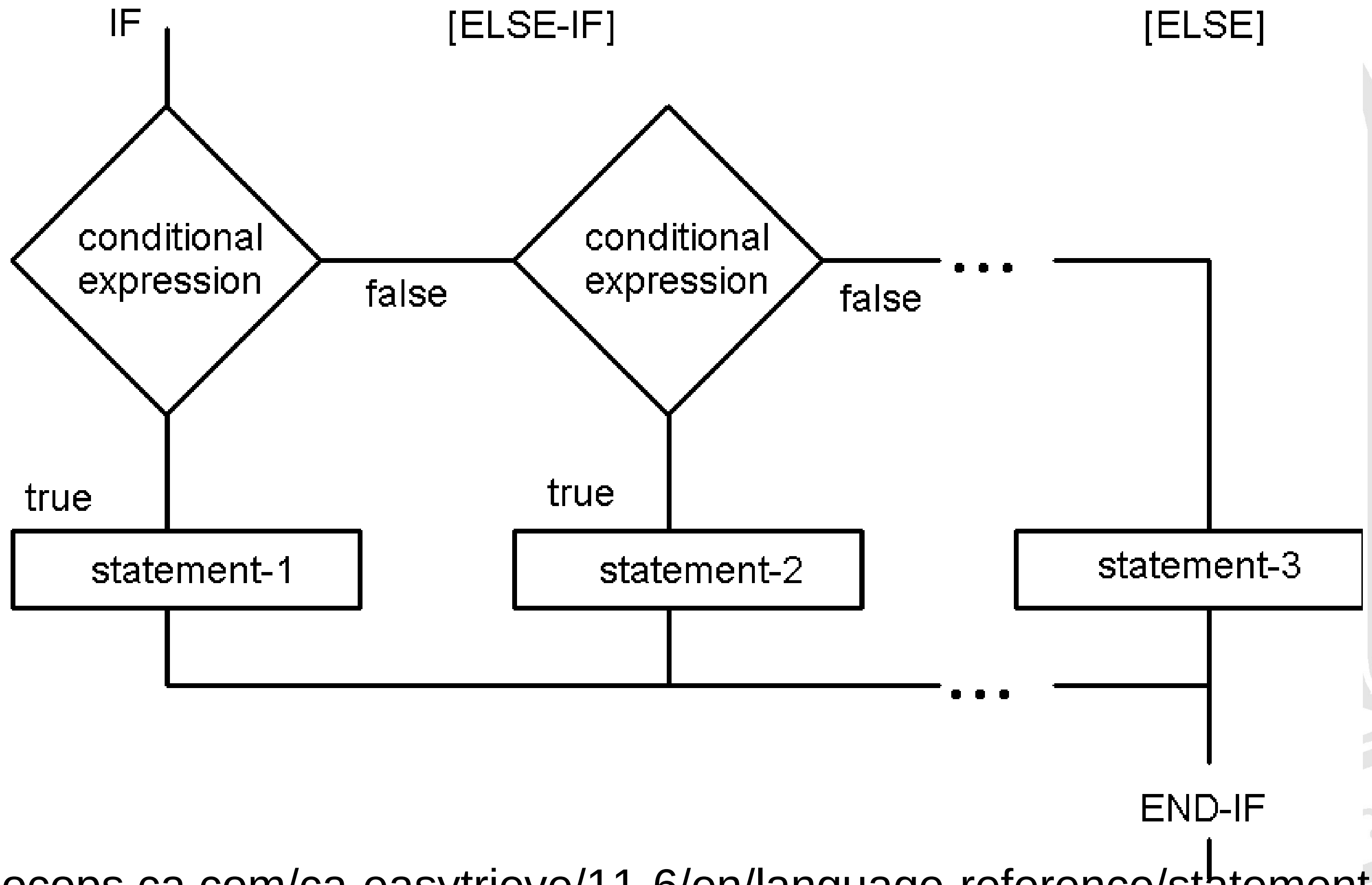
```
A = 6
B = 5
output = 0

if(A > B):
    output = A + B;
else:
    output = A - B;

print('Output: %d'%(output))
```

```
Output: 11
```

Practice

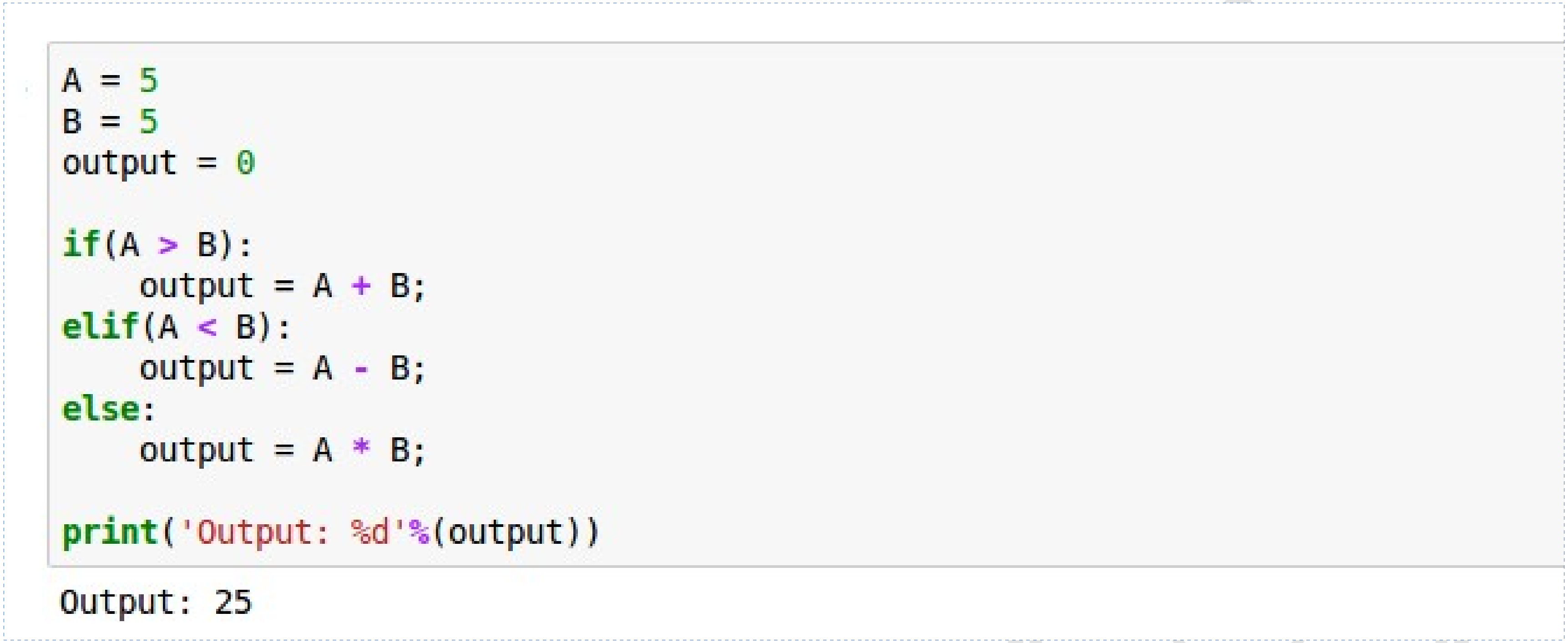
1) เขียนโปรแกรมเพื่อคำนวณเกรด ในกรณีที่
    - นิสิตได้คะแนนตั้งแต่ 0 – 60 จะได้เกรด U
    - นิสิตได้คะแนนตั้งแต่ 61 – 100 จะได้เกรด S

if..elif..else statement



Cr. https://docops.ca.com/ca-easytrieve/11-6/en/language-reference/statements-g-m/if-else-if-else-and-end-if-statements

if..elif..else statement



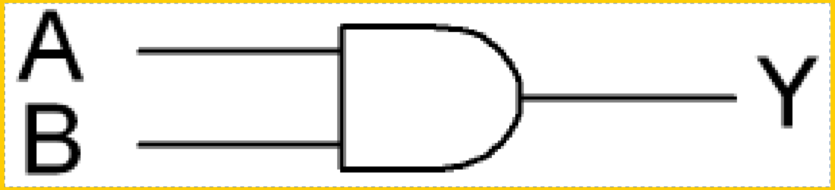
```
A = 5
B = 5
output = 0

if(A > B):
    output = A + B;
elif(A < B):
    output = A - B;
else:
    output = A * B;

print('Output: %d'%(output))
```

```
Output: 25
```

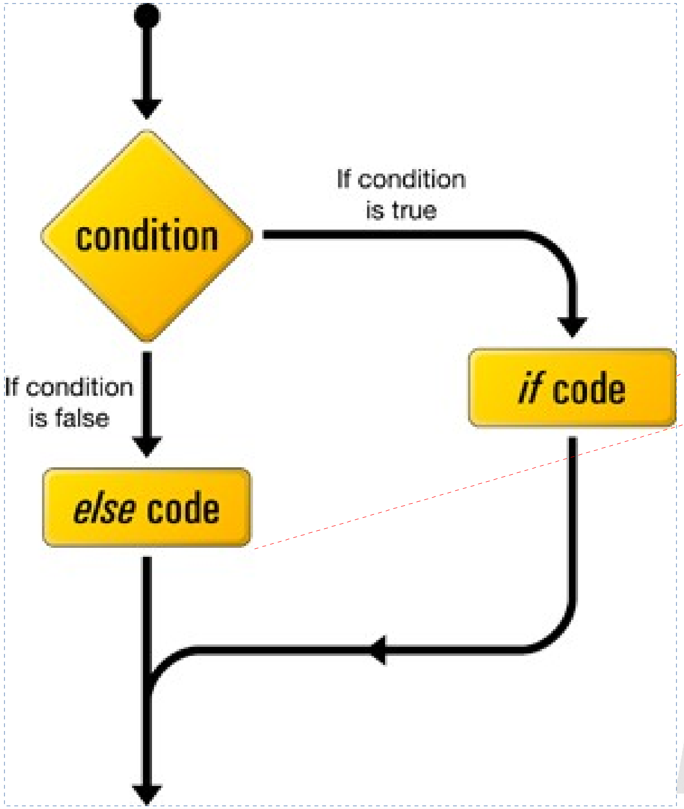
Practice

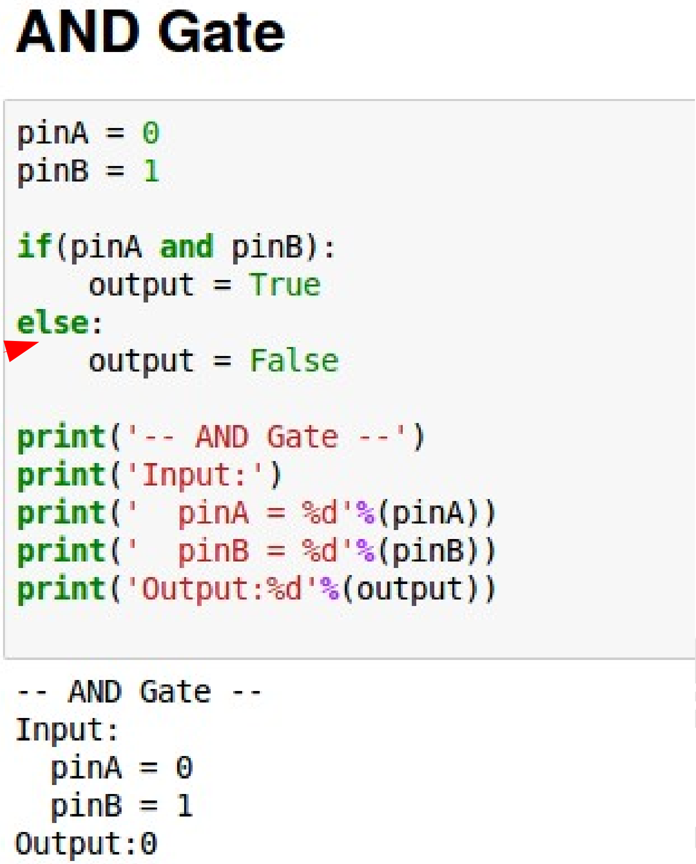
1) เขียนโปรแกรมเพื่อคำนวณเกรด ในกรณีที่
   - นิสิตได้คะแนนตั้งแต่ 0 – 49 จะได้เกรด F
   - นิสิตได้คะแนนตั้งแต่ 50 – 55 จะได้เกรด D+
   - นิสิตได้คะแนนตั้งแต่ 56 – 59 จะได้เกรด D
   - นิสิตได้คะแนนตั้งแต่ 60 – 64 จะได้เกรด C+
   - นิสิตได้คะแนนตั้งแต่ 65 – 69 จะได้เกรด C
   - นิสิตได้คะแนนตั้งแต่ 70 – 74 จะได้เกรด B
   - นิสิตได้คะแนนตั้งแต่ 75 – 79 จะได้เกรด B+
   - นิสิตได้คะแนนตั้งแต่ 80 – 100 จะได้เกรด A

AND gate



- ***if … else*** statement





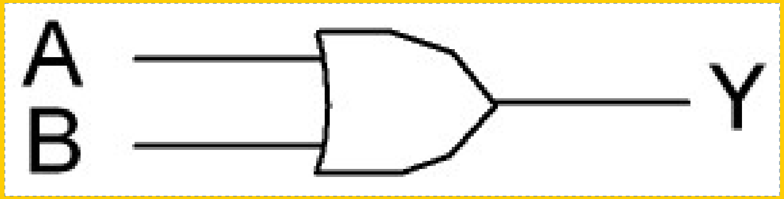
AND Gate

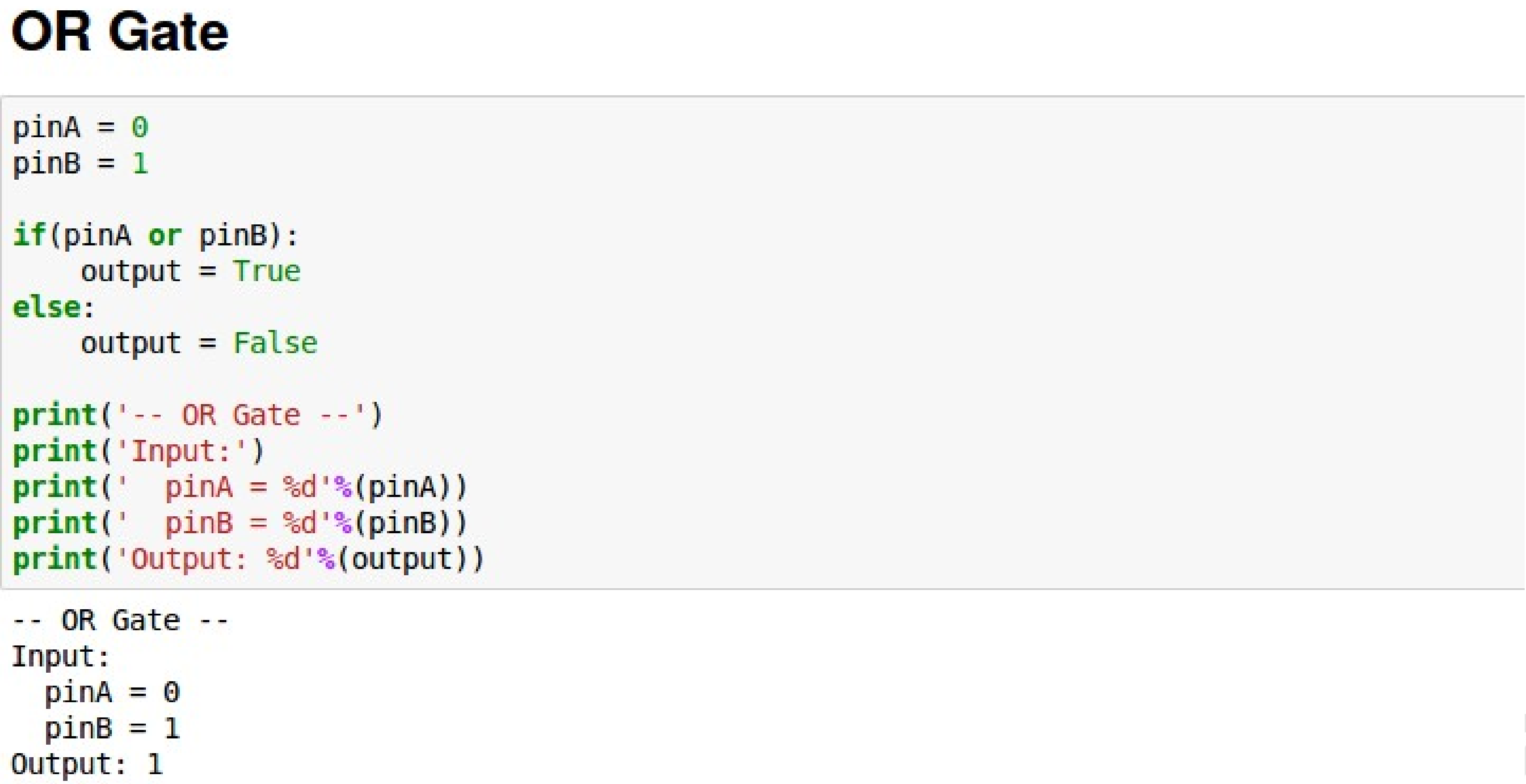
```
pinA = 0
pinB = 1

if(pinA and pinB):
    output = True
else:
    output = False

print('-- AND Gate --')
print('Input:')
print('  pinA = %d'%(pinA))
print('  pinB = %d'%(pinB))
print('Output:%d'%(output))
```

```
-- AND Gate --
Input:
  pinA = 0
  pinB = 1
Output:0
```

Cr. http://phpelearnsite.blogspot.com/2012/07/php-ifelse-statements-conditional.html

OR gate



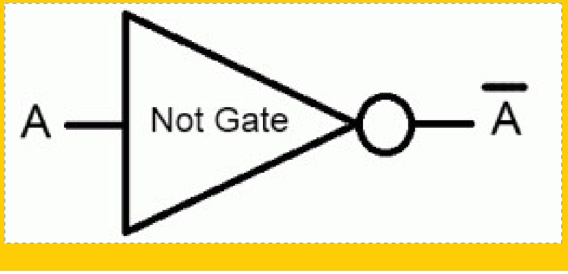


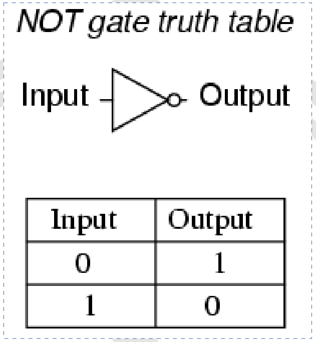
OR Gate

```
pinA = 0
pinB = 1

if(pinA or pinB):
    output = True
else:
    output = False

print('-- OR Gate --')
print('Input:')
print('  pinA = %d'%(pinA))
print('  pinB = %d'%(pinB))
print('Output: %d'%(output))
```

```
-- OR Gate --
Input:
  pinA = 0
  pinB = 1
Output: 1
```

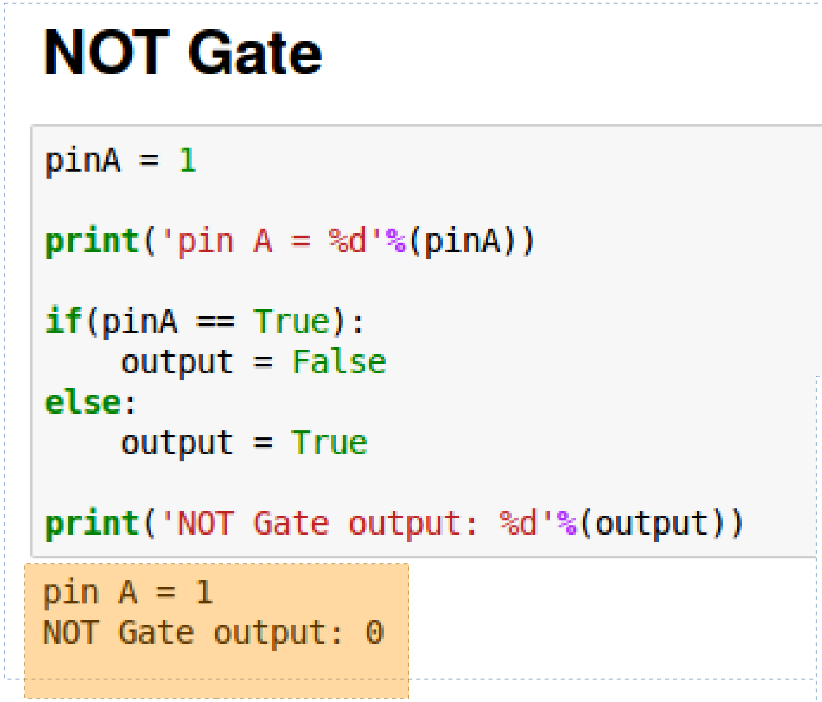
NOT gate




NOT gate truth table

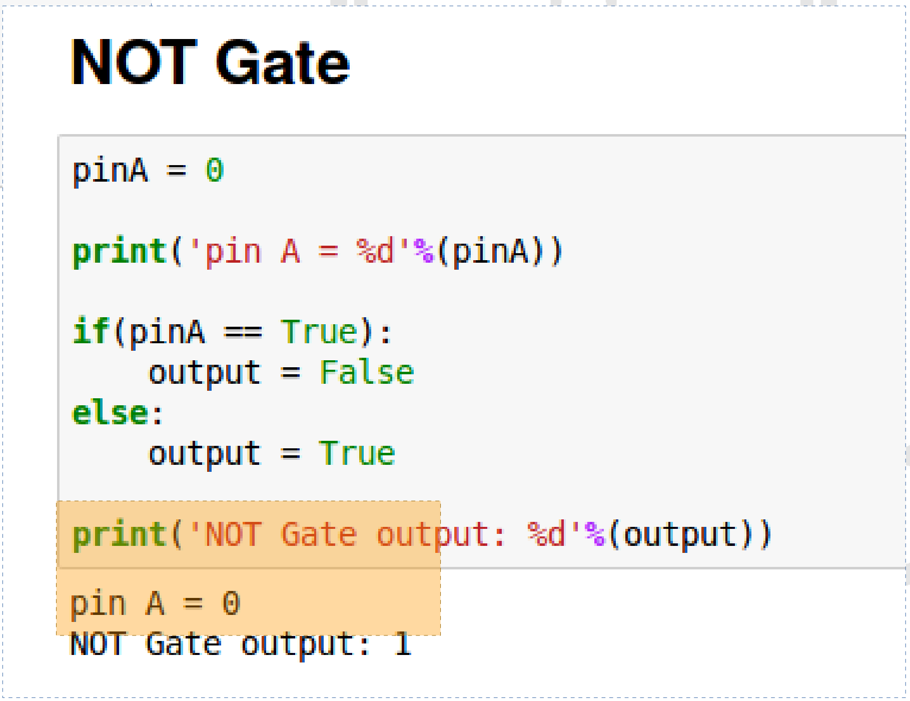
Input
Output

Input	Output
0	1
1	0


NOT Gate

```
pinA = 1

print('pin A = %d'%(pinA))

if(pinA == True):
    output = False
else:
    output = True

print('NOT Gate output: %d'%(output))
```

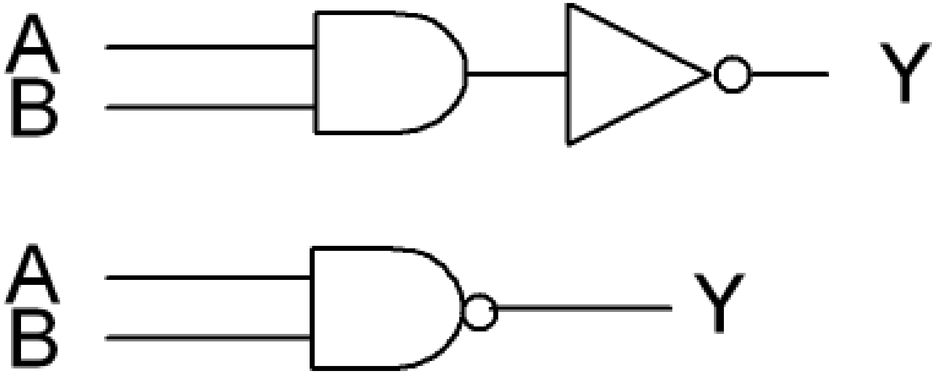
```
pin A = 1
NOT Gate output: 0
```

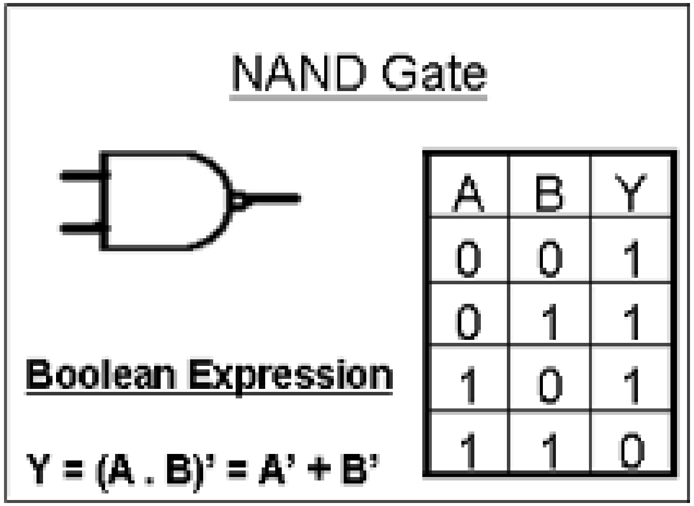

NOT Gate

```
pinA = 0

print('pin A = %d'%(pinA))

if(pinA == True):
    output = False
else:
    output = True

print('NOT Gate output: %d'%(output))
```

```
pin A = 0
NOT Gate output: 1
```

NAND gate





NAND Gate

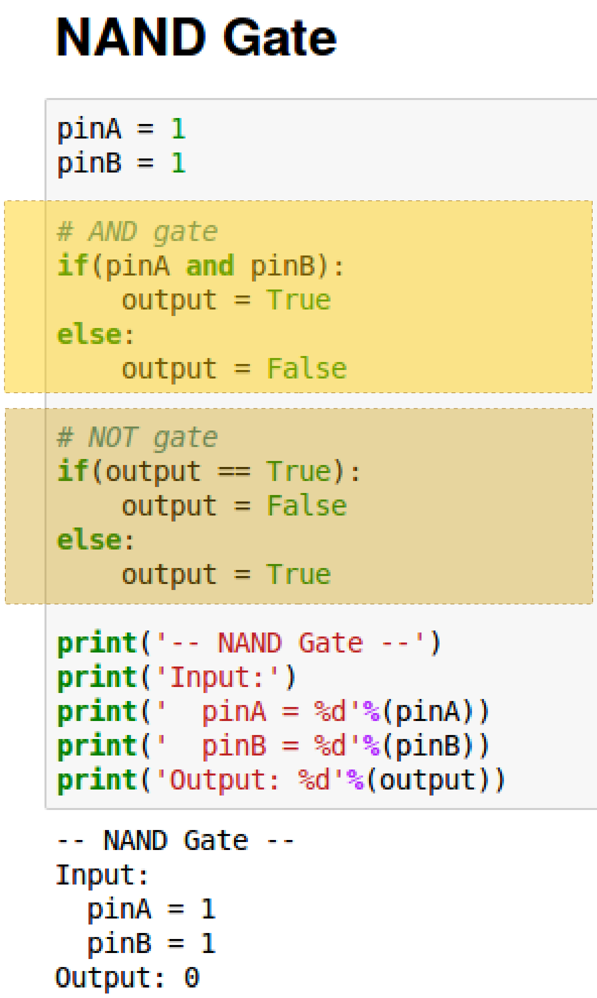
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

Boolean Expression

$Y = (A \cdot B)' = A' + B'$

Cr. http://www.hardwaresecrets.com/introduction-to-logic-gates/4/
http://verticalhorizons.in/universal-gates-in-digital-electonics-nand-and-nor/

NAND gate



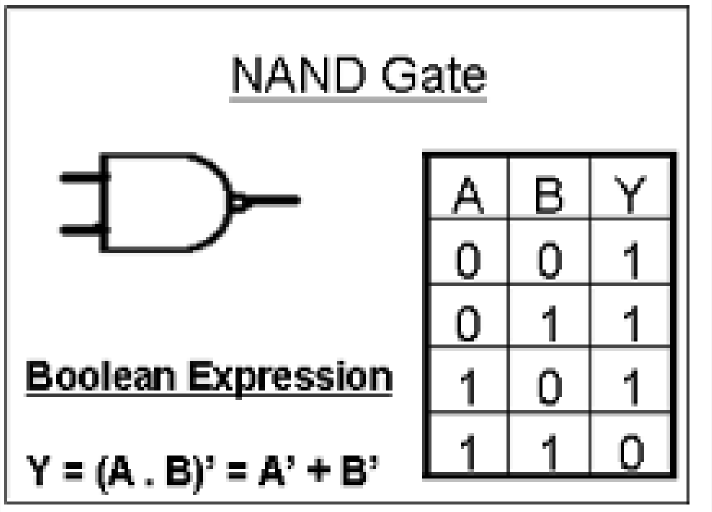
NAND Gate

```
pinA = 1
pinB = 1

# AND gate
if(pinA and pinB):
    output = True
else:
    output = False

# NOT gate
if(output == True):
    output = False
else:
    output = True

print('-- NAND Gate --')
print('Input:')
print('  pinA = %d'%(pinA))
print('  pinB = %d'%(pinB))
print('Output: %d'%(output))
```

```
-- NAND Gate --
Input:
  pinA = 1
  pinB = 1
Output: 0
```

AND gate

NOT gate



NAND Gate

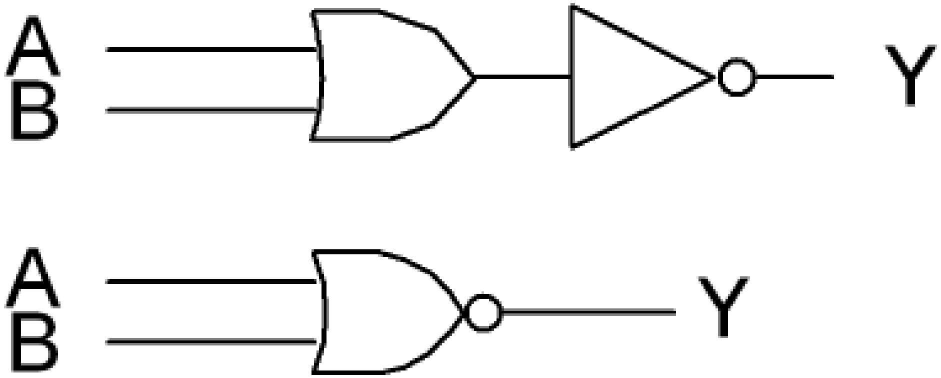
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

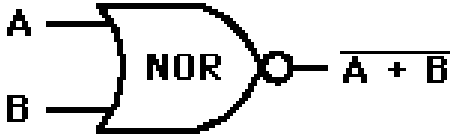
Boolean Expression

Y = (A . B)' = A' + B'

NOR gate

Practice

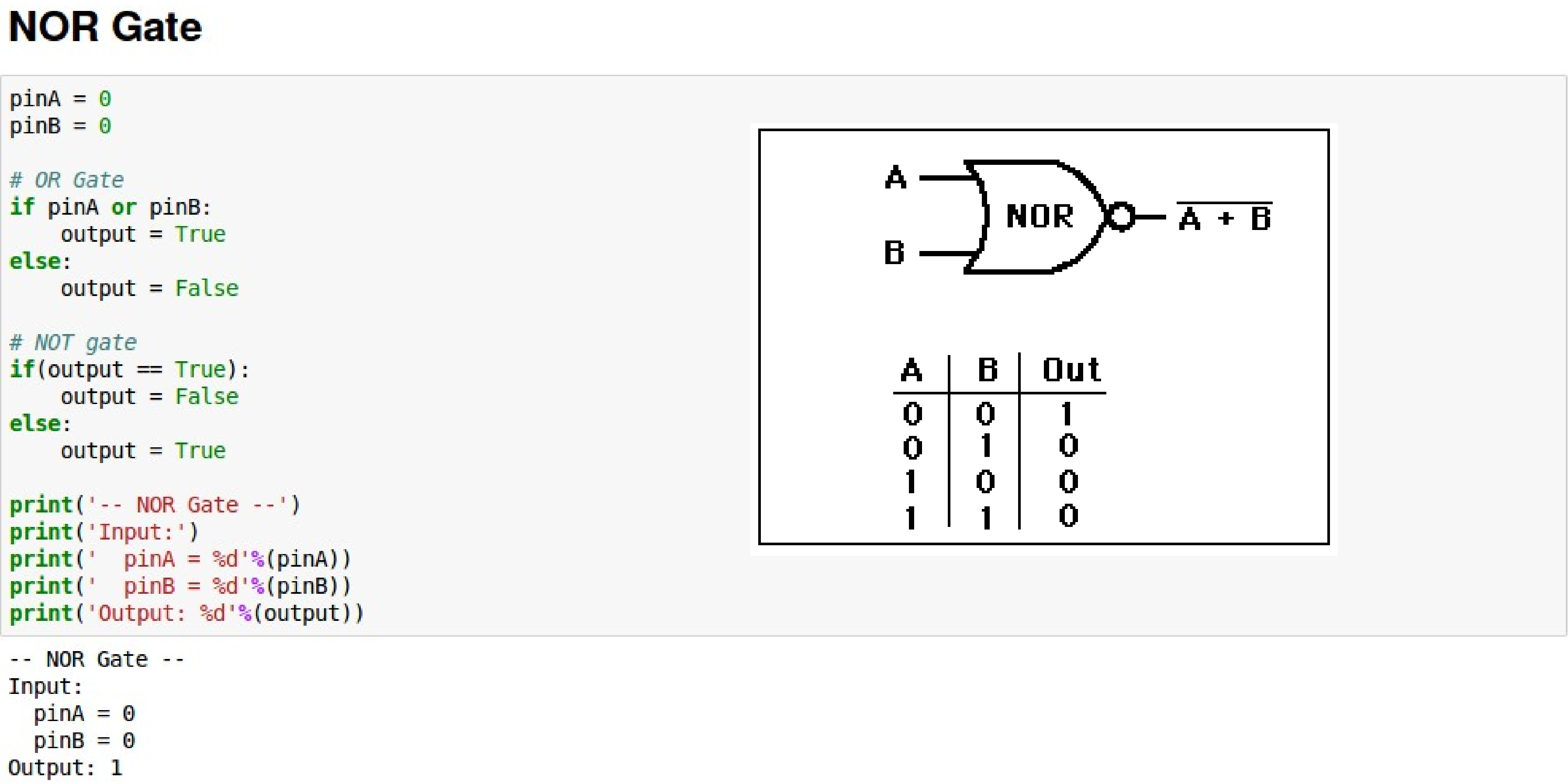




A	B	Out
0	0	1
0	1	0
1	0	0
1	1	0

Cr. http://www.hardwaresecrets.com/introduction-to-logic-gates/4/
http://hyperphysics.phy-astr.gsu.edu/hbase/Electronic/nor.html

NOR gate



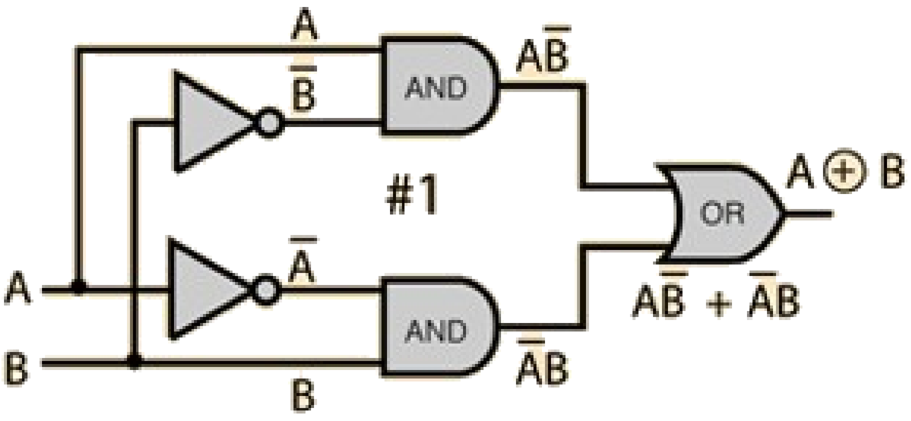
NOR Gate

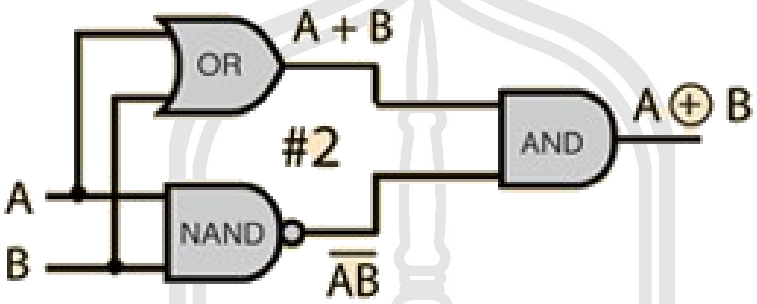
```
pinA = 0
pinB = 0

# OR Gate
if pinA or pinB:
    output = True
else:
    output = False

# NOT gate
if(output == True):
    output = False
else:
    output = True

print('-- NOR Gate --')
print('Input:')
print('  pinA = %d'%(pinA))
print('  pinB = %d'%(pinB))
print('Output: %d'%(output))
```

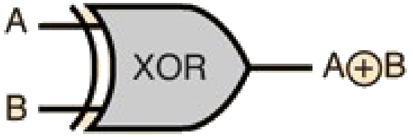
```
-- NOR Gate --
Input:
  pinA = 0
  pinB = 0
Output: 1
```

A	B	Out
0	0	1
0	1	0
1	0	0
1	1	0

XOR gate





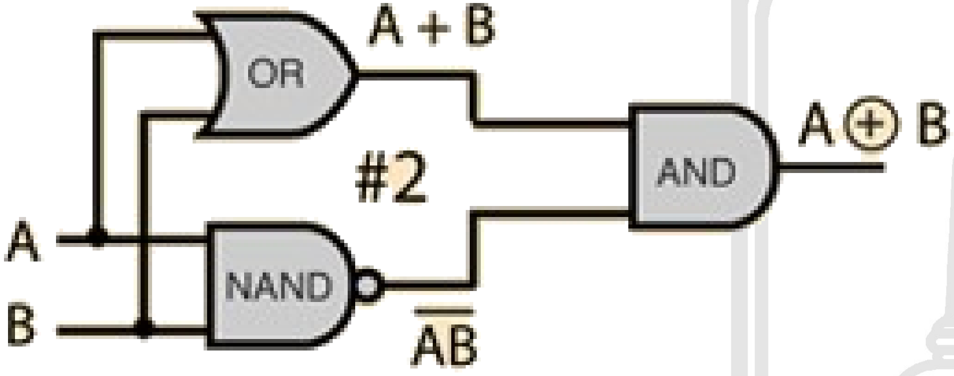


A	B	Out
0	0	0
0	1	1
1	0	1
1	1	0

Cr. http://hyperphysics.phy-astr.gsu.edu/hbase/Electronic/xor.html

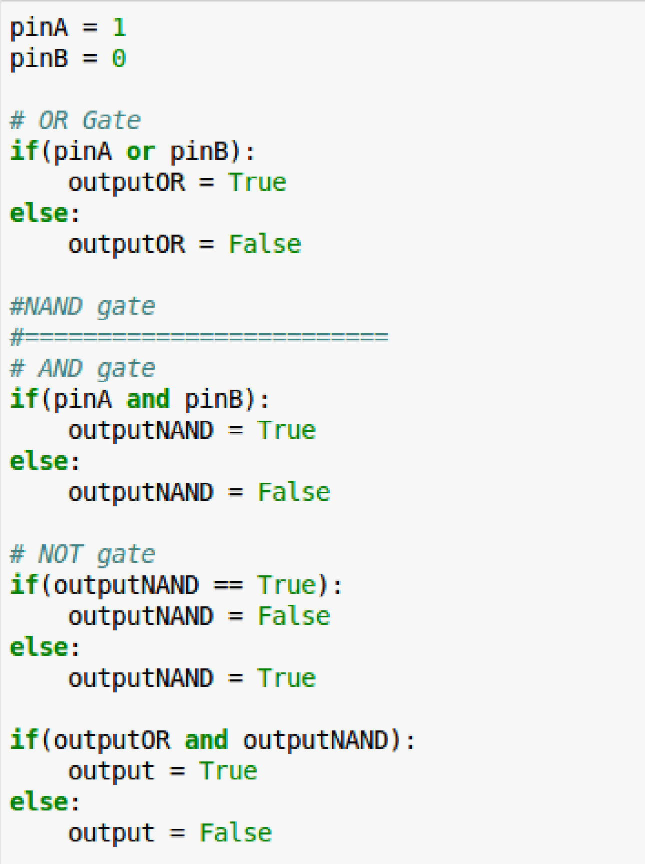
XOR gate

Practice

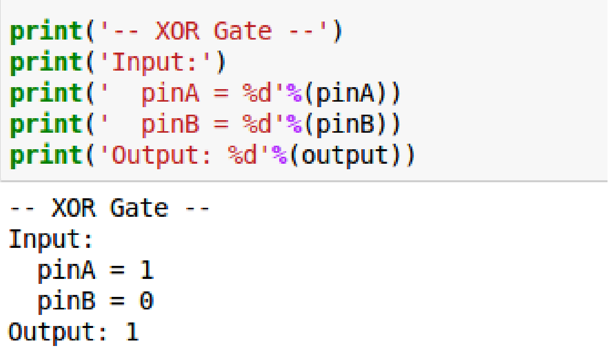


XOR gate

XOR Gate

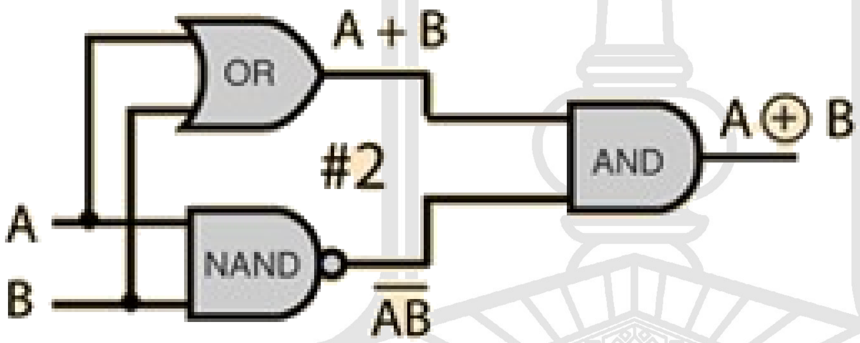


```
pinA = 1
pinB = 0

# OR Gate
if(pinA or pinB):
    outputOR = True
else:
    outputOR = False

#NAND gate
#=========================
# AND gate
if(pinA and pinB):
    outputNAND = True
else:
    outputNAND = False

# NOT gate
if(outputNAND == True):
    outputNAND = False
else:
    outputNAND = True

if(outputOR and outputNAND):
    output = True
else:
    output = False
```



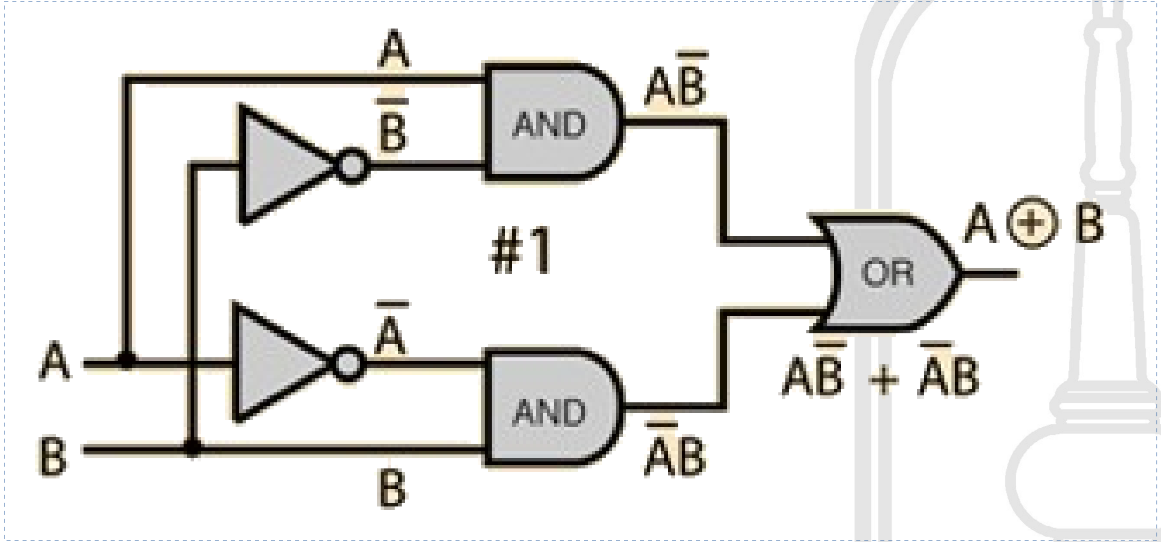
```
print('-- XOR Gate --')
print('Input:')
print('  pinA = %d'%(pinA))
print('  pinB = %d'%(pinB))
print('Output: %d'%(output))
```

```
-- XOR Gate --
Input:
  pinA = 1
  pinB = 0
Output: 1
```



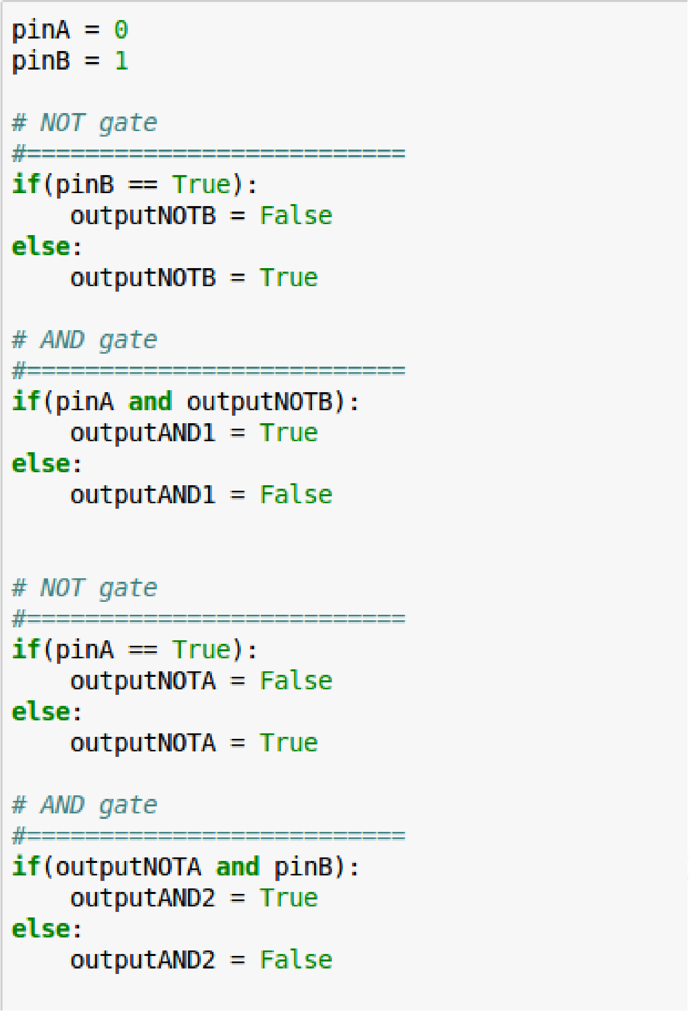
XOR gate

Practice

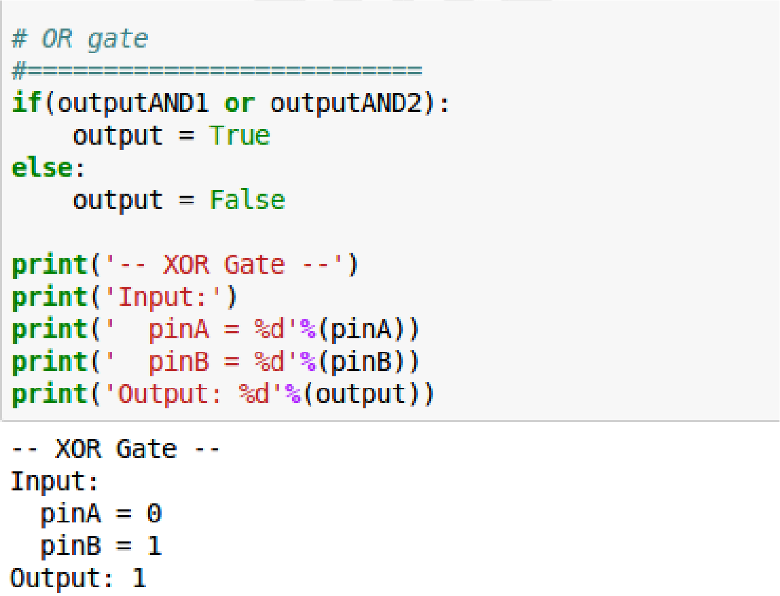


XOR gate

XOR gate

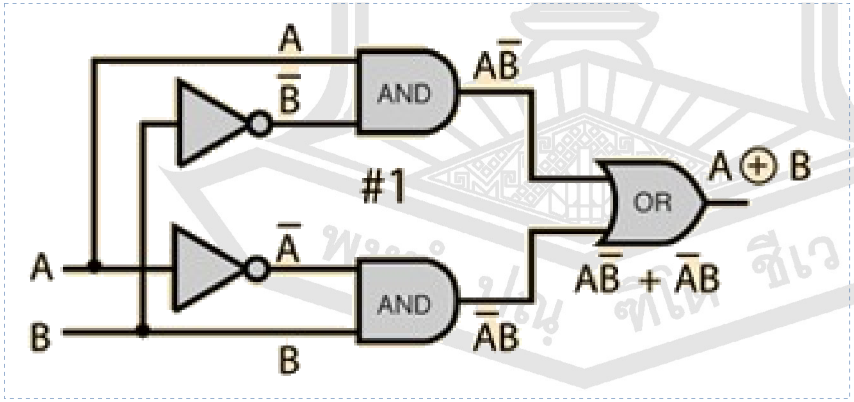


```
pinA = 0
pinB = 1

# NOT gate
#==========================
if(pinB == True):
    outputNOTB = False
else:
    outputNOTB = True

# AND gate
#==========================
if(pinA and outputNOTB):
    outputAND1 = True
else:
    outputAND1 = False


# NOT gate
#==========================
if(pinA == True):
    outputNOTA = False
else:
    outputNOTA = True

# AND gate
#==========================
if(outputNOTA and pinB):
    outputAND2 = True
else:
    outputAND2 = False
```



```
# OR gate
#==========================
if(outputAND1 or outputAND2):
    output = True
else:
    output = False

print('-- XOR Gate --')
print('Input:')
print('  pinA = %d'%(pinA))
print('  pinB = %d'%(pinB))
print('Output: %d'%(output))
```

```
-- XOR Gate --
Input:
  pinA = 0
  pinB = 1
Output: 1
```


A
$\bar{B}$
AND
$A\bar{B}$
#1
OR
$A \oplus B$
$A\bar{B} + \bar{A}B$
$\bar{A}$
AND
$\bar{A}B$
A
B
B