



Unit 5

ຕົວດຳເນີນການທາງຕັກກະສາດ ແລະ ການປຽບທຽບ

(Logic and Comparison Operations)

- Learning objectives
 - ✓ ເພື່ອໃຫ້ເຂົ້າໃຈກ່ຽວກັບຊະນິດຂໍ້ມູນບູລິນ (Boolean) ແລະ ສາມາດປ່ຽນຂໍ້ມູນປະເພດອື່ນໆເປັນຂໍ້ມູນປະເພດບູລິນ.
 - ✓ ເພື່ອໃຫ້ສາມາດປຽບທຽບສອງຂໍ້ມູນທີ່ກ່ຽວຂ້ອງກັນດ້ວຍຕົວດຳເນີນການປຽບທຽບແບບຕ່າງໆ.
 - ✓ ເພື່ອໃຫ້ເຂົ້າໃຈ ແລະ ສາມາດໃຊ້ງານຟັງຊັນ ແລະ ກົດເກນຂອງຕົວດຳເນີນການທາງຕັກສະສາດ.
 - ✓ ເພື່ອໃຫ້ເຂົ້າໃຈ ແລະ ສາມາດທົດລອງການປຽບທຽບຄ່າຕົວເລກ ແລະ ຕົວອັກສອນໂດຍການໃຊ້ຕົວດຳເນີນການປຽບທຽບ ແລະ ຕັກສະສາດ, ກວດສອບຄ່າຜົນຮັບວ່າ ຈິງ ຫຼື ບໍ່ຈິງ ໃນຄ່າຂໍ້ມູນທີ່ເປັນປະເພດບູລິນ.

● Lesson overview

- ✓ ຮຽນຮູ້ກ່ຽວກັບຂໍ້ມູນປະເພດຄ່າຄວາມຈິງ (Boolean data type).
- ✓ ຮຽນຮູ້ກ່ຽວກັບຕົວດຳເນີນການປຽບທຽບ ເຊິ່ງເປັນວິທີການປຽບທຽບຄ່າຂໍ້ມູນໃດໜຶ່ງວ່າ ມີຄ່າໃຫຍ່ກວ່າ, ນ້ອຍກວ່າ ຫຼື ມີຄ່າເທົ່າກັບຄ່າອື່ນໆ.
- ✓ ຮຽນຮູ້ກ່ຽວກັບບົດບາດໜ້າທີ່ ແລະ ການນຳໃຊ້ຕົວດຳເນີນການທາງຕັກກະສາດ.

• Concepts You Will Need to Know From Previous Units

- ✓ ປະເພດຂໍ້ມູນເປັນຕົວເລກຈຳນວນຖ້ວນ, ເລກຈຳນວນຈິງ ແລະ ຂໍ້ມູນປະເພດຂໍ້ຄວາມ ນອກຈາກນັ້ນ ຕ້ອງຮູ້ຈັກ ແລະ ເຂົ້າໃຈກ່ຽວກັບການປະກາດ ແລະ ກຳນົດຄ່າຕົວປ່ຽນ.
- ✓ ສາມາດອ່ານ Code ໂປຣແກຣມ ແລະ ຮູ້ຈັກໃຊ້ງານຕົວດຳເນີນການແບບປະສົມໃນທາງຄະນິດສາດ.
- ✓ ຮູ້ຈັກລຳລັບຄວາມສຳຄັນຂອງຕົວດຳເນີນການປະເພດຕ່າງໆ.

Keywords

Boolean Data Type

Comparison Operator

Logical Operator

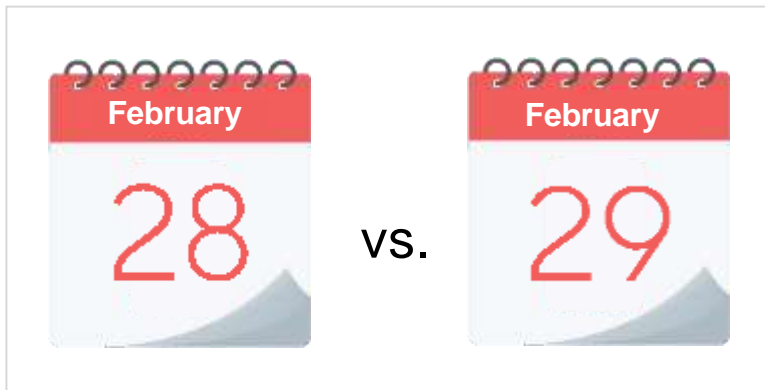
Unit 05

ຕົວດຳເນີນການທາງຕັກກະສາດ ແລະ ການປຸງບາງ

| Mission

1. Real-world Problem

1.1. ຄວາມເປັນມາຂອງປີອະທິກະມາດ



- ໃນໄລຍະເວລາ 1 ປີ, ບໍ່ແນ່ນອນວ່າຈະມີ 365 ວັນ. ມີປີອະທິກະມາດ (leap year) ຈະເປັນປີພິເສດທີ່ເພີ່ມມາອີກ 1 ວັນ, ໃນເດືອນ 2 ຂອງທຸກໆປີຈະມີ 28 ວັນ, ແຕ່ໃນປີອະທິກະມາດ ໃນເດືອນ 2 ຈະມີ 29 ວັນ.
- ປະຕິທິນ Gregorian ແມ່ນປະຕິທິນທີ່ໃຊ້ກັນຢ່າງກວ້າງຂວາງທີ່ສຸດໃນໂລກ, ເປັນການປະຕິຮູບປະຕິທິນ Julian ເຊິ່ງໄດ້ເພີ່ມ 1 ວັນສໍາລັບທຸກໆ 4 ປີ ໃນເດືອນ 2 ໃຫ້ມີ 29 ວັນ.
- ທົດລອງຊອກຫາປີອະທິກະມາດ

1. Real-world Problem

1.2. ຫຼັກການໃນການຄິດໄລ່ຊອກຫາປີອະທິກະມາດ

- ເລກປີໃດຫານຂາດໃຫ້ 4 ຖືວ່າເປັນປີອະທິກະມາດ
- ເລກປີໃດຖ້າຫານຂາດໃຫ້ 4, ຫານບໍ່ຂາດໃຫ້ 100 ຖືວ່າເປັນປີອະທິກະມາດ
- ເລກປີໃດຫານຂາດໃຫ້ 400 ຖືວ່າເປັນປີອະທິກະມາດ.

ຕົວຢ່າງ:

1) $2024 / 4 = 506$

$2024 / 100 = 20.24$ ສະແດງວ່າ ປີ 2024 ເປັນປີອະທິກະມາດ ທີ່ເດືອນ 2 ຈະມີ 29 ວັນ

2) $2000 / 100 = 20$

$2000 / 400 = 5$ ສະແດງວ່າ ປີ 2000 ເປັນປີອະທິກະມາດ ທີ່ເດືອນ 2 ຈະມີ 29 ວັນ

1. Real-world Problem

1.3. ຄິດໄລ່ຫາປີອະທິກະມາດ



The screenshot shows the Omni Calculator website with a red header. The main content area has a light gray background. On the left, there is a sidebar with a menu icon and the text 'omni CALCULATOR'. The main area contains two sections for leap year calculation. The first section is titled 'Single year' and has a text input field labeled 'Your year' and a button labeled 'Is it a leap year?'. The second section is titled 'Range' and has two text input fields labeled 'Start year' and 'End year'.

ເວັບໄຊ:

<https://www.omnicalculator.com/veryday-life/leap-year>

1. Real-world Problem

1.4. ປີອະທິກະມາດ ແລະ ປະຕິທິນເກຣກໍຣຽນ

ກົດເກນການຄິດໄລ່ຫາປີອະທິກະມາດຂອງປະຕິທິນເກຣກໍຣຽນ



<https://www.independent.co.uk/news/uk/home-news/gregorian-calendar-five-facts-a7345126.html>

- ໃນໄລຍະເວລາ 1 ປີ ບໍ່ແນ່ນອນວ່າຈະມີ 365 ວັນ. ມີປີອະທິກະມາດ ຈະເປັນປີພິເສດທີ່ເພີ່ມມາອີກ 1 ວັນ, ໃນເດືອນ 2 ຂອງທຸກໆປີຈະມີ 28 ວັນ, ແຕ່ໃນສໍາລັບປີອະທິກະມາດ ໃນເດືອນ 2 ຈະມີ 29 ວັນ.
- ກົດເກນການຄິດໄລ່ຫາປີອະທິກະມາດສໍາລັບປະຕິທິນເກຣກໍຣຽນ ສະແດງດັ່ງລຸ່ມນີ້:
 - ຕົວເລກຂອງປີ ຄ.ສ ທີ່ສາມາດຫານຂາດໃຫ້ 4 ແມ່ນປີອະທິກະມາດ ຄື: 1984, 1988, 1992, 2004, 2008, 2012, 2016, 2020, 2024, 2028, 2032, 2036, 2040, 2044, 2048, 2052,...
 - ຕົວເລກຂອງປີ ຄ.ສ ທີ່ສາມາດຫານຂາດໃຫ້ 4 ແລະ 100 ຄື: 1700, 1800, 1900, 2100, 2200, 2300,...
 - ຕົວເລກຂອງປີ ຄ.ສ ທີ່ສາມາດຫານຂາດໃຫ້ 4, 100 ແລະ 400 ຄື: 1600, 2000, 2400,...

2. Mission

2.1. ວິທີການຄິດໄລ່ຫາປີອະທິກະມາດ

```
1 year = int(input("Enter the year: "))
2 leap_year = (year % 4 == 0 and year % 100 != 0) or year % 400 == 0
3 print(leap_year)
```

Enter the year: 2021
False

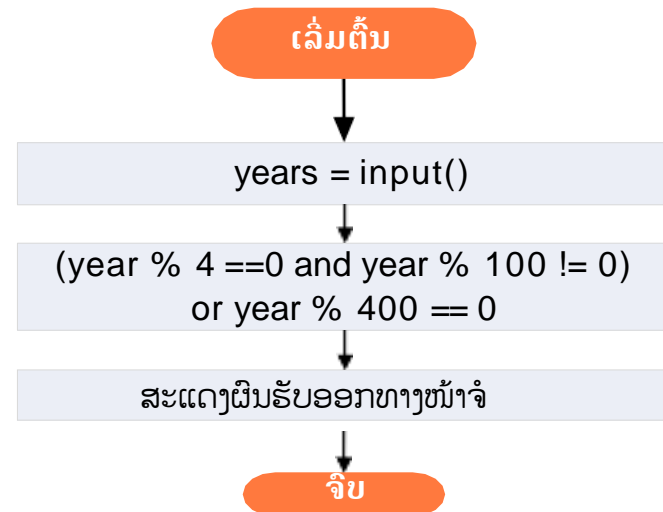
2. Mission

2.2. ແຜນຜັງການຂຽນໂປຣແກຣມ

Pseudocode

1. ເລີ່ມຕົ້ນ
2. ປ້ອນຄ່າເລກປີ
3. ໃຊ້ສູດຄິດໄລ່ຫາປີອະທິກະມາດທີ່ກຳນົດ ເພື່ອໃຫ້ໄດ້ຄ່າ T/F
($\text{year \% } 4 == 0$ and $\text{year \% } 100 != 0$ and $\text{year \% } 400 == 0$)
4. ຖ້າທຸກໆເງື່ອນໄຂເປັນ F(False) ໃຫ້ສະແດງ False
5. ຈົບໂປຣແກຣມ

Flowchart



2. Mission

2.3. ໂປຣແກຣມຄິດໄລ່ຫາປີອະທິກະມາດ

```
1 year = int(input("Enter the year: "))
2 leap_year = (year % 4 == 0 and year % 100 != 0) or year % 400 == 0
3 print(leap_year)
```

ບົດທີ 5 ການດຳເນີນການທາງຕັກກະສາດ ແລະ ການປຽບທຽບ
(Logic and Comparison Operations)

| Key concept

1. ຕົວດຳເນີນການຄ່າຄວາມຈິງ: ຕົວດຳເນີນການ Logical ແລະ ປຽບທຽບ

1.1. ຕົວດຳເນີນການປຽບທຽບ

ຕົວດຳເນີນການປຽບທຽບ (Comparison Operator) ຈະໃຫ້ຜົນຮັບຂອງການປຽບທຽບຄ່າຂໍ້ມູນທີ່ເປັນຕົວເລກວ່າ ຄ່າໃດໃຫຍ່ກວ່າ ຫຼື ນ້ອຍກວ່າ ຈາກນັ້ນຈະສົ່ງຄ່າກັບມາເປັນຄ່າຄວາມຈິງ 2 ຄ່າຄື: ຈິງ (True) ຫຼື ບໍ່ຈິງ (False).



1. ຕົວດຳເນີນການຄຳຄວາມຈິງ: ຕົວດຳເນີນການ Logical ແລະ ປຽບທຽບ

1.1. ຕົວດຳເນີນການປຽບທຽບ

- ຕົວດຳເນີນການປຽບທຽບ (Comparison Operator) ຈະໃຫ້ຜົນຮັບຂອງການປຽບທຽບຄ່າຂໍ້ມູນທີ່ເປັນຕົວເລກວ່າ ຄ່າໃດໃຫຍ່ກວ່າ ຫຼື ນ້ອຍກວ່າ.
- ຈາກນັ້ນຈະສົ່ງຄ່າກັບມາເປັນຄ່າຄວາມຈິງ 2 ຄ່າ ຄື: ຈິງ (True) ຫຼື ບໍ່ຈິງ (False) ປະເພດຂໍ້ມູນທີ່ມີຄ່າ True ຫຼື False ເອີ້ນວ່າ ຄ່າຄວາມຈິງ ຫຼື Booleans.

```
1 print('Result of 10 > 5: ', 10 > 5)
2 print('Result of 5 < 1: ', 5 > 1)
3 print('Result of 5 == 5: ', 5 == 5)
4 print('Result of 5 != 5: ', 5 != 5)
5 print('Result of 'a' > 'b': ', 'a' > 'b')
```

```
Result of 10 > 5: True
Result of 5 < 1: False
Result of 5 == 5: True
Result of 5 != 5: False
Result of 'a' > 'b': False
```

1. ຕົວດຳເນີນການຄ່າຄວາມຈິງ: ຕົວດຳເນີນການ logical ແລະ ປຽບທຽບ

1.2. ປະເພດຂໍ້ມູນຄ່າຄວາມຈິງ

- **Boolean:** ແມ່ນປະເພດຂໍ້ມູນທີ່ປະກອບດ້ວຍ 2 ຄ່າ ຄື: ຈິງ (True) ແລະ ບໍ່ຈິງ (False)
- ສາມາດເວົ້າໄດ້ວ່າເປັນປະເພດຂໍ້ມູນແບບ Logical ແລະ ນຳໃຊ້ໃນການສະແດງຄ່າ ຈິງ ແລະ ບໍ່ຈິງ, ສັງເກດຕົວຢ່າງລຸ່ມນີ້ ໃນພາສາ Python ຕົວເລກຈຳນວນຖ້ວນ 1 ສາມາດປ່ຽນເປັນຄ່າ True ແລະ ເລກ 0 ສາມາດປ່ຽນເປັນ False

```
1 print(bool(1))
2 print(bool(0))
3 print(bool(True))
4 print(bool(False))
```

```
True
False
True
False
```




One More Step

ທົດລອງສະແດງຄ່າຂໍ້ມູນຊະນິດຄ່າຄວາມຈິງ

```
1 print(bool(-1))
2 print(bool(0))
3 print(bool(None))
4 print(bool(''))
5 print(bool('hello'))
6 print(bool([]))
7 print(bool([10, 20]))
```

True
False
False
False
True
False
True

- ເລີ່ມຕົ້ນ ເລກ 0 ເວລາປ່ຽນເປັນຊະນິດຄ່າຄວາມຈິງຈະໃຫ້ຄ່າເປັນ False . ຕົວເລກອື່ນໆຍົກເວັ້ນ 0 ເວລາປ່ຽນເປັນຊະນິດຄ່າຄວາມຈິງຈະໃຫ້ຄ່າອອກມາເປັນ True
- ການຍະຫວ່າງ ຈະໃຫ້ຄ່າອອກມາເປັນ False, ສ່ວນຕົວອັກສອນອື່ນໆຮ່ວມກັບການຍະຫວ່າງຈະໃຫ້ຄ່າອອກມາເປັນ True.
- ເຄື່ອງໝາຍ [] ທີ່ເປັນ object ຖ້າບໍ່ມີຄ່າພາຍໃນກໍຈະໃຫ້ຄ່າອອກມາເປັນ False.

1. ຕົວດຳເນີນການຄ່າຄວາມຈິງ: ຕົວດຳເນີນການ logical ແລະ ປຽບທຽບ

1.3. ອະທິບາຍກ່ຽວກັບຕົວດຳເນີນການປຽບທຽບ

ຕົວດຳເນີນການ	ຄຳອະທິບາຍ	a = 100 b = 200
==	ຖ້າສອງຕົວຖືກດຳເນີນການມີຄ່າຄືກັນຈະໃຫ້ຜົນຮັບອອກມາເປັນ True	a == b is False
!=	ຖ້າສອງຕົວຖືກດຳເນີນການມີຄ່າຄືກັນຈະໃຫ້ຜົນຮັບອອກມາເປັນ False	a != b is True
>	ຖ້າຕົວຖືກດຳເນີນການຢູ່ເບື້ອງຊ້າຍໃຫຍ່ກວ່າ ຕົວຖືກດຳເນີນການຢູ່ເບື້ອງຂວາຈະໃຫ້ຄ່າເປັນ True	a > b is False
<	ຖ້າຕົວຖືກດຳເນີນການຢູ່ເບື້ອງຊ້າຍນ້ອຍກວ່າ ຕົວຖືກດຳເນີນການຢູ່ເບື້ອງຂວາຈະໃຫ້ຄ່າເປັນ True	a < b is True
>=	ຖ້າຕົວຖືກດຳເນີນການຢູ່ເບື້ອງຊ້າຍໃຫຍ່ກວ່າ ຫຼື ເທົ່າກັບ ຕົວຖືກດຳເນີນການຢູ່ເບື້ອງຂວາຈະໃຫ້ຄ່າເປັນ True	a >= b is False
<=	ຖ້າຕົວຖືກດຳເນີນການຢູ່ເບື້ອງຊ້າຍນ້ອຍກວ່າ ຫຼື ເທົ່າກັບ ຕົວຖືກດຳເນີນການຢູ່ເບື້ອງຂວາຈະໃຫ້ຄ່າເປັນ True	a <= b is True

1. ຕົວດຳເນີນການຄ່າຄວາມຈິງ: ຕົວດຳເນີນການ logical ແລະ ປຽບທຽບ

1.4. ໂປຣແກຣມຕົວຢ່າງຂອງການໃຊ້ຕົວດຳເນີນການປຽບທຽບ

 ການໃຫ້ຄ່າເປັນຄ່າຄວາມຈິງ ໃນເວລາໃຊ້ຕົວດຳເນີນການປຽບທຽບສະແດງດັ່ງຕົວຢ່າງລຸ່ມນີ້:

```
1 a = 100
2 b = 200
3 print(a == b)
4 print(a != b)
5 print(a > b)
6 print(a < b)
7 print(a >= b)
8 print(a <= b)
```

```
False
True
False
True
False
True
```

- ຕົວດຳເນີນການ == ຈະໃຫ້ຄ່າເປັນ ຈິງ (True) ກໍຕໍ່ເມື່ອສອງພາກມີຄ່າຄືກັນເທົ່ານັ້ນ ເຊັ່ນ: ໃຫ້ a == b ຖ້າ a = 100 ແລະ b = 200 ກໍຈະໃຫ້ຄ່າເປັນ False
- ຕົວດຳເນີນການ != ຈະໃຫ້ຄ່າເປັນ ຈິງ (True) ກໍຕໍ່ເມື່ອສອງພາກມີຄ່າຕ່າງກັນເທົ່ານັ້ນ ເຊັ່ນ: ໃຫ້ a != b ຖ້າ, a=100 ແລະ b = 200 ກໍຈະໃຫ້ຄ່າເປັນ True.
- ດ້ວຍເຫດນີ້ ຕົວດຳເນີນການປຽບທຽບຈະປຽບທຽບຄ່າຂໍ້ມູນໃນລັກສະນະໃຫຍ່ກວ່າ, ນ້ອຍກວ່າ, ເທົ່າກັບ, ໃຫຍ່ກວ່າ ຫຼື ເທົ່າກັບ ນ້ອຍກວ່າ ຫຼື ເທົ່າກັບ.

1. ຕົວດຳເນີນການຄຳຄວາມຈິງ: ຕົວດຳເນີນການ logical ແລະ ປຽບທຽບ

1.4. ໂປຣແກຣມຕົວຢ່າງຂອງການໃຊ້ຕົວດຳເນີນການປຽບທຽບ

- ການໃຫ້ຄ່າຂອງຕົວດຳເນີນການປຽບທຽບ

```
1 n = int(input('Enter the integer: '))  
2 print('Is the integer even?', n % 2 == 0)
```

```
Enter the integer: 60  
Is the integer even? True
```

ແຖວທີ 2

- ຕ້ອງການຫານໃຫ້ 2 ເພື່ອເອົາຄ່າເສດ ຖ້າເສດອອກເປັນ 0 ຈະໃຫ້ຄ່າເປັນ True ແຕ່ຖ້າບໍ່ແມ່ນຈະໃຫ້ຄ່າເປັນ False .

1. ຕົວດຳເນີນການຄຳຄວາມຈິງ: ຕົວດຳເນີນການ logical ແລະ ປຽບທຽບ

1.5. ຂໍ້ຄວາມ ແລະ ຕົວດຳເນີນການປຽບທຽບ

- ໃນຂໍ້ຄວາມໜຶ່ງ, ຕົວດຳເນີນການ == ຈະໃຫ້ຄ່າ True ຖ້າວ່າ ຂໍ້ຄວາມທັງສອງເບື້ອງຄືກັນ

```
1 print('aaa' == 'aaa')
2 print('aaa' == 'bbb')
3 print('aaa' != 'aaa')
4 print('aaa' != 'bbb')
```

True
False
False
True

ແຖວທີ 1-2

- ໃນແຖວທີ 1 ໃຫ້ຄ່າເປັນ True ຍ້ອນວ່າ ຂໍ້ຄວາມທັງສອງເບື້ອງມີຄ່າຄືກັນ.
- ໃນແຖວທີ 2 ໃຫ້ຄ່າເປັນ False ຍ້ອນວ່າ ຂໍ້ຄວາມທັງສອງເບື້ອງມີຄ່າຕ່າງກັນ ຫຼື ບໍ່ເທົ່າກັນ.

1. ຕົວດຳເນີນການຄ່າຄວາມຈິງ: ຕົວດຳເນີນການ logical ແລະ ປຽບທຽບ

1.5. ຂໍ້ຄວາມ ແລະ ຕົວດຳເນີນການປຽບທຽບ

- ໃນຂໍ້ຄວາມໜຶ່ງ, ຕົວດຳເນີນການ == ຈະໃຫ້ຄ່າ True ຖ້າວ່າ ຂໍ້ຄວາມທັງສອງເບື້ອງຄືກັນ

```
1 print('aaa' == 'aaa')
2 print('aaa' == 'bbb')
3 print('aaa' != 'aaa')
4 print('aaa' != 'bbb')
```

```
True
False
False
True
```



ແຖວທີ 3-4

- ໃນແຖວທີ 3 ໃຫ້ຄ່າເປັນ False ຍ້ອນວ່າ ຂໍ້ຄວາມທັງສອງເບື້ອງຄືກັນ.
- ໃນແຖວທີ 4 ໃຫ້ຄ່າເປັນ True ຍ້ອນວ່າ ຂໍ້ຄວາມທັງສອງເບື້ອງບໍ່ຄືກັນ.

1. ຕົວດຳເນີນການຄຳຄວາມຈິງ: ຕົວດຳເນີນການ logical ແລະ ປຽບທຽບ

1.6. ຄວາມແຕກຕ່າງລະຫວ່າງ == ແລະ is

- ໃນຄຳສັ່ງ **is** ຈະກວດເບິ່ງວ່າ object ທັງສອງຄືກັນ ຫຼື ບໍ່

```
1 a = 1
2 b = 1.0
3 print(a == b)
4 print(a is b)
```

True
False

📌 ແຖວທີ 3-4

- ໃນແຖວທີ 3 ໃຊ້ຕົວດຳເນີນການ == ເພື່ອກວດເບິ່ງຄ່າຂໍ້ມູນສອງຄ່າຄືກັນ ຫຼື ບໍ່, ຖ້າຂໍ້ມູນທັງສອງມີຄ່າຄືກັນກໍຈະໃຫ້ຄ່າເປັນ True.
- ໃນແຖວທີ 4 ໃຊ້ is ເພື່ອກວດເບິ່ງວ່າ object ທັງສອງວ່າຄືກັນ ຫຼື ບໍ່, ເຊິ່ງ a ເກັບຄ່າເປັນເລກຈຳນວນຖ້ວນ, ສ່ວນ b ເກັບຄ່າເປັນເລກຈຳນວນຈິງ ດັ່ງນັ້ນທັງສອງ object ມີຄ່າຊະນິດຕ່າງກັນ ສະນັ້ນຈະໃຫ້ຄ່າເປັນ False.



One More Step

- ຕົວດຳເນີນການ **in** ເພື່ອກວດເບິ່ງຂໍ້ມູນບາງສ່ວນຂອງຂໍ້ມູນມີສ່ວນທີ່ຄືກັນ ຫຼື ບໍ່

```
1 print('aaa' in 'aaa-bbb-ccc')
2 print('bbb' in 'aaa-bbb-ccc')
3 print('ddd' in 'aaa-bbb-ccc')
```

True
True
False

- ຖ້າຕ້ອງການກວດເບິ່ງວ່າມີບາງສ່ວນຂໍ້ຄວາມໃນຂໍ້ຄວາມ ເຮົາສາມາດໃຊ້ຕົວດຳເນີນ in ເພື່ອໃຫ້ໄດ້ຄ່າ True ຫຼື False.
- ຈາກຕົວຢ່າງໂປຣແກຣມຂ້າງເທິງ ໃນຄະນະທີ່ aaa ແມ່ນມີຢູ່ໃນຂໍ້ຄວາມ aaa-bbb-ccc ດັ່ງນັ້ນໂປຣແກຣມຈະໃຫ້ຄ່າເປັນ True ກັບມາ.



One More Step

- ພາສາ Python ສາມາດໃຊ້ສອງ ຫຼື ຫຼາຍຕົວດຳເນີນການປຽບທຽບລວມກັນໄດ້ ດັ່ງຕົວຢ່າງລຸ່ມນີ້.
- ໃນໂປຣແກຣມລຸ່ມນີ້, ເມື່ອ $n = 100$, $0 < n < 200$ ຈະໃຫ້ຄ່າເປັນ True. ຄຳສັ່ງ **if** ຈະກວດເບິ່ງ ເງື່ອນໄຂເພື່ອປະມວນຜົນໃນເວລາເງື່ອນໄຂເປັນ True, ເຊິ່ງພວກເຮົາຈະໄດ້ຮຽນໃນ Unit ຕໍ່ໄປ.

```
1 n = 100
2 print('n =', n)
3 if 0 < n < 200 :
4     print('n is between 0 and 200.')
```

n = 100
n is between 0 and 200.

1. ຕົວດຳເນີນການຄ່າຄວາມຈິງ: ຕົວດຳເນີນການ logical ແລະ ປຽບທຽບ
- 1.7. ຂໍ້ຄວນລະວັງໃນເວລາໃຊ້ຕົວດຳເນີນການປຽບທຽບ

⚙️ ຂໍ້ຜິດພາດສາມາດເກີດຂຶ້ນໃນເວລາລຳດັບການດຳເນີນການມີການປ່ຽນແປງ

```
1 num1 = 100
2 num2 = 200
3 num1 =< num2
```

```
File "<ipython-input-34-efe2ac59bee3>", line 3
num1 =< num2
      ^
```

SyntaxError: invalid syntax

- ຕົວດຳເນີນການ <= ບໍ່ແມ່ນ =<
- ຫ້າມມີການຍະຫວ່າງ ລະຫວ່າງ ! ແລະ = ຫຼື > ກັບ =

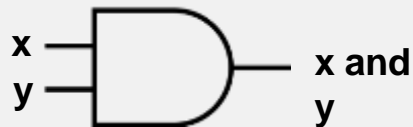
2. ຕົວດຳເນີນການຕັກກະສາດ: ການກຳນົດສຳນວນທາງຕັກກະສາດ

2.1. ຕົວດຳເນີນການທາງຕັກສະສາດ: and , or , not

ການກຳນົດນິພົດທາງຕັກກະສາດ ແລະ ໃຫ້ຄ່າ True ແລະ False.

x and y

x	y	x and y
False	False	False
False	True	False
True	False	False
True	True	True



x or y

x	y	x or y
False	False	False
False	True	True
True	False	True
True	True	True



not x

x	not x
False	True
True	False




ຮູບສະແດງຂ້າງເທິງນີ້ ແມ່ນສັນຍະລັກລຳຈິກເກດທີ່ໃຊ້ໃນທາງວິສະວະກຳອີເລັກໂຕຣນິກ ແລະ ວິທະຍາສາດຄອມພິວເຕີ; ແຕ່ລະສັນຍະລັກມີຄວາມໝາຍວ່າ: and, or, not ຕາມລຳດັບ

2. ຕົວດຳເນີນການຕັກກະສາດ: ການກຳນົດສຳນວນທາງຕັກກະສາດ

2.1. ຕົວດຳເນີນການທາງຕັກສະສາດ: and , or , not

ຕົວດຳເນີນການ	ຄຳອະທິບາຍ
x and y	ຖ້າໜຶ່ງໃນ x ຫຼື y ມີຄ່າເປັນ False ຈະໃຫ້ຄ່າເປັນ False ແລະ ຈະໃຫ້ຄ່າເປັນ True ກໍລະນີທັງສອງມີຄ່າເປັນ True
x or y	ຖ້າໜຶ່ງໃນ x ຫຼື y ມີຄ່າເປັນ True ຈະໃຫ້ຄ່າເປັນ True ແລະ ຈະໃຫ້ຄ່າເປັນ False ກໍລະນີທັງສອງມີຄ່າເປັນ False
not x	ຖ້າ x ເປັນ True ຈະໃຫ້ຄ່າຜົນຮັບເປັນ False, ຖ້າ x ເປັນ False ຈະໃຫ້ຄ່າຜົນຮັບເປັນ True.

 ໃນເວລາຕົວດຳເນີນການຖືກນຳໃຊ້ງານຈະໃຫ້ຄ່າເປັນຊະນິດຄ່າຄວາມຈິງ

```
1 x = True
2 y = False
3 print(x and y)
4 print(x or y)
5 print(not x)
6 print(not y)
```

```
False
True
False
True
```

2. ຕົວດຳເນີນການຕັກກະສາດ: ການກຳນົດທາງຕັກກະສາດ

2.1. ຕົວດຳເນີນການທາງຕັກສະສາດ: and , or , not

ຕົວດຳເນີນການທາງຕັກກະສາດຈະລວມຄ່າຄວາມຈິງເຂົ້າດ້ວຍກັນເພື່ອສ້າງຄ່າຄວາມຈິງໃໝ່ຂຶ້ນມາ

```
1 num = int(input('Enter an Integer: '))
2 result = ( num >= 0 and num <= 100 and num % 2 == 0 )
3 print('Is the input an even integer between 0 and 100?', result)
```

Enter an Integer: 99

Is the input an even integer between 0 and 100? False

ແຖວທີ 2

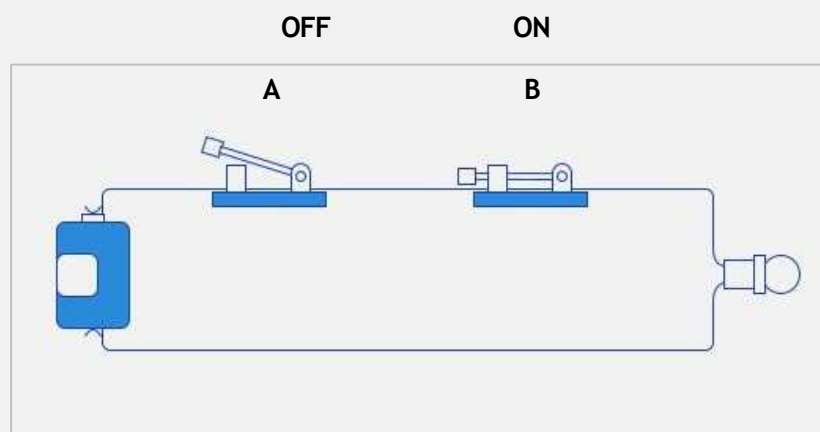
- ຕົວປ່ຽນ num ມີຄ່າໃຫຍ່ກວ່າເທົ່າກັບ 0 , ແຕ່ນ້ອຍກວ່າເທົ່າກັບ 100, ສະນັ້ນສົມຜົນທີ 1 ແລະ 2 ຈະໃຫ້ຄ່າເປັນ True.
- ໃນສົມຜົນທີ 3 ຕົວປ່ຽນ $num \% 2 = 1$, ສະນັ້ນ $1 == 0$ ຈະໃຫ້ຄ່າເປັນ False.
- ການດຳເນີນການທາງຕັກກະສາດຈະໃຫ້ຄ່າເປັນ True ເມື່ອທຸກໆສົມຜົນເປັນ True. ດັ່ງນັ້ນໃນໂປຣແກຣມຕົວຢ່າງຂ້າງເທິງຈະໃຫ້ຄ່າເປັນ False.

2. ຕົວດຳເນີນການຕັກກະສາດ: ການກຳນົດສຳນວນທາງຕັກກະສາດ

2.2. ປຽບທຽບຕົວດຳເນີນການ and ກັບ or

ໃນເວລານຳໃຊ້ຕົວດຳເນີນການ and ຈະໃຫ້ຄ່າຜົນຮັບເປັນ False ຈາກຄ່າອື່ນພຸດບາງຄ່າເປັນ False. ຈະໃຫ້ຄ່າເປັນ True ກໍ່ຕໍ່ເມື່ອອື່ນພຸດທັງສອງມີຄ່າເປັນ True

ຖ້າສະວິດໃດໜຶ່ງປິດ ຈະເຮັດໃຫ້ດອກໄຟບໍ່ຮຸ່ງ (False)

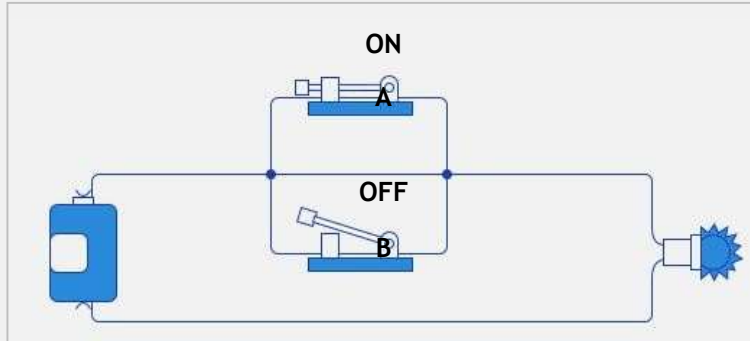


Input		Output
A	B	
True	True	True
True	False	False
False	True	False
False	False	False

2. ຕົວຕຳເນີນການຕັກກະສາດ: ການກຳນົດນິພົດທາງຕັກກະສາດ

2.2. ປຽບທຽບຕົວຕຳເນີນການ and ກັບ or

ໃນເວລານຳໃຊ້ຕົວຕຳເນີນການ **or** ຈະໃຫ້ຄ່າຜົນຮັບເປັນ False ຈາກຄ່າອື່ນພຸດທີ່ມີຄ່າເປັນ False ໝົດທຸກຕົວ. ຈະໃຫ້ຄ່າເປັນ True ກໍຕໍ່ເມື່ອອື່ນພຸດບາງຕົວມີຄ່າເປັນ True ຖ້າສະວິດໃດໜຶ່ງເປີດ ຈະເຮັດໃຫ້ດອກໄຟຮຸ່ງ (True)



or

Input		Output
A	B	
True	True	True
True	False	True
False	True	True
False	False	False

Unit 5 ການດຳເນີນການທາງຕັກກະສາດ ແລະ ການປຽບທຽບ (Logic and Comparison Operations)

Paper Coding

- ພະຍາຍາມທຳຄວາມເຂົ້າໃຈພື້ນຖານແນວຄວາມຄິດກ່ອນຈະກ້າວໄປຍັງຂັ້ນຕອນຖັດໄປ.
- ຖ້າບໍ່ເຂົ້າໃຈພື້ນຖານແນວຄວາມຄິດກໍ່ຈະພາໃຫ້ມີບັນຫາໃນການຮຽນສຳລັບຫຼັກສູດນີ້ ອາດຈະເຮັດໃຫ້ເຮົາບໍ່ຜ່ານຫຼັກສູດນີ້.
- ມັນອາດຈະຂ້ອນຂ້າງຍາກໃນຕອນນີ້, ແຕ່ຜົນສຳເລັດຂອງຫຼັກສູດນີ້ພວກເຮົາຂໍແນະນຳໃຫ້ທ່ານເຂົ້າໃຈຢ່າງລະອຽດກ່ຽວກັບແນວຄວາມຄິດ ແລ້ວຈຶ່ງກ້າວໄປຂັ້ນຕອນຕໍ່ໄປ.

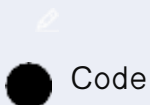
Q1. ຈົ່ງຂຽນໂປຣແກຣມຮັບຄ່າ n ຈາກຜູ້ໃຊ້ຜ່ານແປ້ນພິມ ຈະໄດ້ຄ່າຜົນຮັບເປັນ True ເມື່ອ n ຈຳນວນຖ້ວນຄືກ
ແລະ ໃຫ້ຄ່າຜົນຮັບເປັນ False ເມື່ອ n ທີ່ຮັບເຂົ້າມາເປັນເລກຖ້ວນຄູ່.
ສໍາລັບການທົດລອງແມ່ນໃຫ້ n ມີຄ່າເປັນ 20 ແລະ 21

ເງື່ອນໄຂສໍາລັບ
ການປະມວນຜົນ

Enter an Integer : 20
Is the integer odd?: :
False

ໃຊ້ເວລາ

Enter an Integer : 21
Is the integer odd?: True
5 ນາທີ



Q2. ຈົ່ງຂຽນ Code ທີ່ມີການຮັບຄ່າຈາກຜູ້ໃຊ້ ແລະ ກຳນົດວ່າຄ່າຈຳນວນຖ້ວນ n ເປັນເລກຄູ່ ທີ່ຢູ່ລະຫວ່າງ 0-100 ຫຼື ບໍ່. ຜົນໄດ້ຮັບຂອງການປະມວນຜົນສັງເກດດັ່ງລຸ່ມນີ້:

ເງື່ອນໄຂສໍາລັບ ການປະມວນຜົນ	Enter an integer: 12 Is the input an even integer between 0 and 100?
ໃຊ້ເວລາ	True 5 ນາທີ



Code

ບົດທີ 5 ການດຳເນີນການທາງຕັກກະສາດ ແລະ ການປຽບທຽບ (Logic and Comparison Operations)

| Let's Code

1. ຕົວດຳເນີນການຕັກກະສາດ: AND, OR , NOT

1.1. AND, OR , NOT ໃຫ້ຄ່າເປັນ True/False

- ພາສາ Python ສະໜັບສະໜູນຕົວດຳເນີນການ AND, OR, NOT ທີ່ຮູ້ຈັກກັນທົ່ວໄປວ່າເປັນຕົວດຳເນີນການທາງຕັກກະສາດ. ການດຳເນີນການເຫຼົ່ານີ້ຈະໃຫ້ຄ່າເປັນຄ່າຄວາມຈິງ True ຫຼື False ດ້ວຍຕົວດຳເນີນການເຫຼົ່ານີ້.
- ຂໍ້ມູນຄ່າຄວາມຈິງເປັນປະເພດຂໍ້ມູນທີ່ມີ 2 ຄ່າ ຄື: True ແລະ False, ຊະນິດຂໍ້ມູນອື່ນໆທີ່ບໍ່ແມ່ນຄ່າຄວາມຈິງສາມາດປ່ຽນເປັນຊະນິດຄ່າຄວາມຈິງໄດ້.
- ກ່ອນອື່ນໝົດ, ເລກ 0 ທີ່ຖືກປ່ຽນແລ້ວ ແລະ ໃຫ້ຄ່າເປັນ False ສໍາລັບຄ່າຄວາມຈິງ. ສ່ວນຕົວເລກອື່ນໆໃກ້ຄຽງ 0 ທີ່ຖືກປ່ຽນຈະໃຫ້ຄ່າເປັນ True, ຕົວຢ່າງ: ຖ້າ $a = 0$, $b = 1$ ຈາກນັ້ນເອົາ a AND b ຈະໄດ້ຜົນອອກມາເປັນ False
- ຂໍ້ມູນທີ່ເປັນຍະຫວ່າງຈະຖືກແປງ ແລະ ໃຫ້ຄ່າເປັນ False ສໍາລັບຄ່າຄວາມຈິງ ໃນຂະນະທີ່ຕົວອັກສອນທັງໝົດຈະໃຫ້ຄ່າເປັນ True.
- ສໍາລັບຂໍ້ມູນທີ່ບໍ່ມີຄ່າ ທີ່ຖືກແປງແລ້ວຈະໃຫ້ຄ່າເປັນ False.

1. ຕົວດຳເນີນການຕັກກະສາດ: AND, OR , NOT

1.1. AND, OR, NOT ໃຫ້ຄ່າເປັນ True/False

```
1 print(10 > 20) # 10 ນ້ອຍກວ່າ 20, ສະນັ້ນ 10 > 20 ຈະໃຫ້ຄ່າເປັນ False
2 print(10 < 20) # 10 ນ້ອຍກວ່າ 20, ສະນັ້ນ 10 < 20 ຈະໃຫ້ຄ່າເປັນ True
3 print(bool(9)) # 9 ບໍ່ແມ່ນ 0, ສະນັ້ນຈະໃຫ້ຄ່າເປັນ True
4 print(bool(-1)) # -1 ບໍ່ແມ່ນ 0, ສະນັ້ນຈະໃຫ້ຄ່າເປັນ True
5 print(bool(0)) # 0 ແມ່ນຄ່າຕົວເລກເທົ່ານັ້ນ ສະແດງວ່າໃຫ້ຄ່າເປັນ False
6 print(bool(None)) # None ມີຄວາມໝາຍວ່າບໍ່ມີຄ່າ ສະນັ້ນໃຫ້ຄ່າເປັນ False
7 print(bool('')) # ຍະຫວ່າງແມ່ນຫຼັງຕົວອັກສອນ ສະນັ້ນໃຫ້ຄ່າເປັນ False
8 print(bool('hello')) # ແມ່ນຄ່າຂໍ້ມູນທີ່ເປັນຂໍ້ຄວາມ ສະນັ້ນໃຫ້ຄ່າເປັນ True
9
```

False
True
True
True
False
False
False
True

1. ຕົວດຳເນີນການຕັກກະສາດ: AND, OR , NOT

1.2. AND, OR , NOT ໃຫ້ຄ່າເປັນ True/False

- ຈາກຜົນຮັບຂອງບົດຝຶກຫັດທີ່ຜ່ານມາ ສັງເກດເຫັນວ່າ 9 ແລະ - 1 ໃຫ້ຄ່າຄວາມຈິງເປັນ True, ແຕ່ 0 ແລະ None ໃຫ້ຄ່າຄວາມຈິງເປັນ False.
- ໃນທີ່ນີ້ none ແມ່ນຄຳສະຫງວນໃນພາສາ Python ສະແດງວ່າມັນບໍ່ມີຄ່າ. ເພື່ອຄວາມແນ່ໃຈເຮົາມາກວດເບິ່ງນຳກັນວ່າ 0 ແລະ ຍະຫວ່າງແມ່ນຈະໃຫ້ຄ່າເປັນ False ແລະ ຄ່າອື່ນໆທີ່ໃຫ້ຄ່າເປັນ True.

1. ຕົວດຳເນີນການຕັກກະສາດ: AND, OR , NOT

1.3. AND, OR , NOT ໃຫ້ຄ່າເປັນ True/False

- ການດຳເນີນການສາມາດນຳໃຊ້ກັບຂໍ້ມູນຊະນິດຄ່າຄວາມຈິງ ວິທີດັ່ງກ່າວເປັນການດຳເນີນການທາງຕັກກະສາດ. ການດຳເນີນທາງຕັກກະສາດລວມເອົາຄ່າຄວາມຈິງຫຼາຍຄ່າເພື່ອສ້າງຄ່າຄວາມຈິງໃໝ່ຂຶ້ນມາ

ຕົວດຳເນີນການທາງຕັກກະສາດໃນພາສາ Python

ຕົວດຳເນີນການ	ຄຳອະທິບາຍ
x and y	ຖ້າຄ່າໃດໜຶ່ງຂອງ x ຫຼື y ເປັນ False ຈະໃຫ້ຜົນຮັບເປັນ False. ຈະໃຫ້ຄ່າເປັນ True ໃນກໍລະນີ x ແລະ y ມີຄ່າເປັນ True ທັງສອງ.
x or y	ຖ້າຄ່າໃດໜຶ່ງຂອງ x ຫຼື y ເປັນ True ຈະໃຫ້ຜົນຮັບເປັນ True. ຈະໃຫ້ຄ່າເປັນ False ໃນກໍລະນີ x ແລະ y ມີຄ່າເປັນ False ທັງສອງ.
not x	ຖ້າ x ມີຄ່າເປັນ True ຈະໃຫ້ຜົນຮັບອອກມາເປັນ False, ຖ້າ x ມີຄ່າເປັນ False ກໍ່ຈະໃຫ້ຄ່າເປັນ True

1. ຕົວດຳເນີນການຕັກກະສາດ: AND, OR , NOT

1.4. ຕາຕະລາງຄ່າຄວາມຈິງ

ຄ່າຜົນຮັບທີ່ໄດ້ຈະຂຶ້ນກັບຕົວດຳເນີນການທາງຕັກກະສາດທີ່ດຳເນີນການກັບຄ່າຂອງ x ແລະ y ເຊິ່ງສະແດງດັ່ງຕາຕະລາງລຸ່ມນີ້:

x and y

x	y	x and y
False	False	False
False	True	False
True	False	False
True	True	True

x or y

x	y	x OR Y
False	False	False
False	True	True
True	False	True
True	True	True

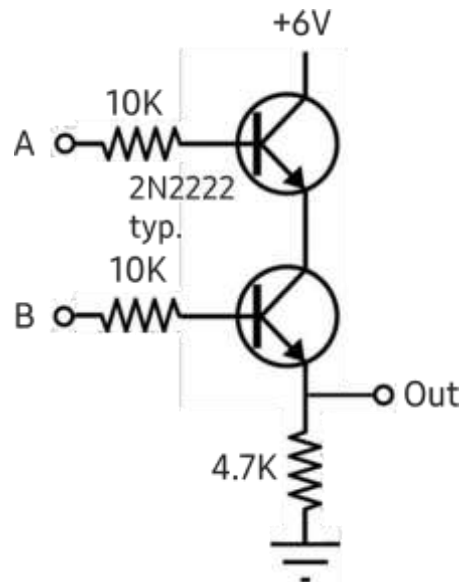
not x

x	not x
False	True
True	False



One More Step

ວິຈານ Logic AND, OR



Symbol		Truth Table		
		B	A	Q
0		0	0	0
0		1	0	0
1		0	0	0
1		1	1	1
		B	A	Q
0		0	0	0
0		1	1	1
1		0	1	1
1		1	1	1

2. ລຳດັບການດຳເນີນການ

2.1. ລຳດັບຄວາມສຳຄັນຂອງຕົວດຳເນີນການໃນພາສາ Python

- ພວກເຮົານຳໃຊ້ຫຼາຍຕົວດຳເນີນການໃນເວລາຂຽນໂປຣແກຣມ ແລະ ລຳດັບຄວາມສຳຄັນໃນເວລາທີ່ນຳມາລວມກັນເປັນສິ່ງສຳຄັນ.

$$\begin{array}{c}
 x + y \times z \\
 \underbrace{\hspace{1.5cm}} \textcircled{1} \\
 \underbrace{\hspace{2.5cm}} \textcircled{2}
 \end{array}$$

$$\begin{array}{c}
 (x + y) \times z \\
 \textcircled{1} \underbrace{\hspace{1.5cm}} \\
 \textcircled{2} \underbrace{\hspace{2.5cm}}
 \end{array}$$

- ຄືດັ່ງທີ່ເຮົາຮຽນໃນວິຊາຄະນິດສາດ, ການຄູນ ແລະ ການຫານຄວນດຳເນີນການກ່ອນການບວກ ແລະ ການລົບ. ການກຳນົດລຳດັບຄວາມສຳຄັນກໍຄືລຳດັບຂອງການດຳເນີນການ.
- ນີ້ແມ່ນກົດເກນໃນການກຳນົດວ່າຕົວດຳເນີນການໃດໃນຫຼາຍໆການດຳເນີນການຈະຖືກປະຕິບັດກ່ອນ. ຕົວດຳເນີນການແຕ່ລະຕົວຈະຖືກລຽງລຳດັບ, ການຄູນ ແລະ ການຫານຈະມີລຳດັບຄວາມສຳຄັນສູງກວ່າການບວກ ແລະ ການລົບ.

2. ລໍາດັບການດໍາເນີນການ

2.2. ຕາຕະລາງລໍາດັບຄວາມສໍາຄັນຂອງຕົວດໍາເນີນການ

ຕົວດໍາເນີນການ	ຄໍາອະທິບາຍ
**	ຕົວດໍາເນີນການຂຶ້ນກຳລັງ
~ , + , -	ຕົວດໍາເນີນການເພີ່ມຄ່າ ຫຼື ຫຼຸດຄ່າ
* , / , % , //	ຕົວດໍາເນີນການຄູນ, ຫານ, ຫານເອົາຄ່າເສດ, ຫານເອົາຄ່າຖ້ວນຂອງຜົນຫານ
+ , -	ຕົວດໍາເນີນການບວກ, ລົບ
>> , <<	ຕົວດໍາເນີນການໃນລະດັບບິດຂອງຂໍ້ມູນ
&	ຕົວດໍາເນີນການໃນລະດັບບິດຂອງ AND
^ ,	ຕົວດໍາເນີນການໃນລະດັບບິດຂອງ XOR, OR
<= , < , > , >=	ຕົວດໍາເນີນການປຽບທຽບ
== , !=	ຕົວດໍາເນີນການທຽບເທົ່າ
= , %= , /= , //= , -= , += , *= , **=	ຕົວດໍາເນີນການກຳນົດຄ່າ
is , is not	ຕົວດໍາເນີນການປຽບທຽບ Object
in , not in	ຕົວດໍາເນີນການກວດເບິ່ງການບັນຈຸໃນຂໍ້ມູນ
not , or , and	ຕົວດໍາເນີນການທາງຕັກກະສາດ

ຕົວດໍາເນີນການຢູ່ຕໍາແໜ່ງເທິງສຸດຂອງຕາຕະລາງແມ່ນມີລໍາດັບຄວາມສໍາຄັນຫຼາຍກວ່າໝູ່ຕາມລໍາດັບ

2. ລໍາດັບຄວາມສໍາຄັນຂອງຕົວດໍາເນີນການ

2.3. ຕົວຢ່າງໂປຣແກຣມລໍາດັບຄວາມສໍາຄັນຂອງຕົວດໍາເນີນການ

```
1 x = int(input("Enter the First Equation: "))
2 y = int(input("Enter the Second Equation: "))
3 z = int(input("Enter the Third Equation: "))
4
5 avg = (x + y + z) / 3
6 print("Average =", avg)
```

Enter the First Equation: 10

Enter the Second Equation: 20

Enter the Third Equation: 30

Average = 20.0



ແຖວທີ 5

- ຖ້າພວກເຮົາຂຽນ $x + y + z/3$ ໂປຣແກຣມກໍຈະດໍາເນີນການໃນສ່ວນ $z/3$ ກ່ອນ. ດັ່ງນັ້ນ, ພວກເຮົາຕ້ອງການດໍາເນີນການບວກກ່ອນກໍຈະຂຽນໄວ້ໃນເຄື່ອງໝາຍວົງເລັບ ເຊັ່ນ: $(x+y+z)/3$
- ຄ້າຍຄືກັບສູດທີ່ໃຊ້ໃນວິຊາຄະນິດສາດ, ຄ່າທີ່ຢູ່ພາຍໃນເຄື່ອງໝາຍວົງເລັບຈະມີລໍາດັບຄວາມສໍາຄັນສູງກວ່າ.

ບົດທີ 5 ການດຳເນີນການທາງຕັກກະສາດ ແລະ ການປຽບທຽບ
(Logic and Comparison Operations)

| Pair Programming



Pair programming Plactice

ການຂຽນໂປຣແກຣມແບບເປັນຄູ່ ເປັນການຂຽນໂປຣແກຣມທີ່ໃຫ້ສອງຄົນເຮັດວຽກໃນງານດຽວກັນ ພ້ອມກັນໂດຍໃຊ້ຄອມພິວເຕີໜ່ວຍດຽວ ແຕ່ໃຊ້ສະໝອງສອງຄົນຊ່ວຍກັນຄິດ ເຊິ່ງອາດຈະໃຫ້ຄົນທີ່ເກັ່ງກວ່າເປັນຄົນເບິ່ງ ຄົນທີ່ອ່ອນກວ່າ(ໃນກໍລະນີທີ່ຄວາມສາມາດຂອງສອງຄົນຕ່າງກັນ) ແຕ່ໃນກໍລະນີຄົນສອງຄົນຄວາມສາມາດໃກ້ຄຽງກັນ ກໍ່ອາດຈະໃຫ້ຄົນໃດຄົນໜຶ່ງເປັນຄົນຂຽນ ແລະ ອີກຄົນໃຫ້ເປັນຜູ້ຕິດຕາມ,ແນະນຳ ຖ້າມີຄຳຜິດກໍ່ບອກຄົນຂຽນເພື່ອແກ້ໄຂໃຫ້ໂປຣແກຣມທີ່ຂຽນນັ້ນຖືກຕ້ອງ.

- ແນວທາງການຂຽນໂປຣແກຣມແບບນີ້ໃຊ້ໄດ້ຫຼາຍກໍລະນີ ເຊັ່ນ:
 - ✓ ຖ້າຕ້ອງຂຽນໂປຣແກຣມຍາກໆທີ່ຄິດຄົນດຽວບໍ່ອອກ ກໍ່ລອງຫາຄູ່ຂຽນໂປຣແກຣມທີ່ມີຄວາມສາມາດໃນລະດັບດຽວກັນ ຫຼື ໃຫ້ຄຽງກັນ ເພື່ອພັດປ່ຽນການຂຽນໂປຣແກຣມ ແລະ ທຳງານ.
 - ✓ ຕ້ອງການພັດທະນານັກຂຽນໂປຣແກຣມມືໃໝ່ ເພື່ອຮັບເຂົ້າທີມພັດທະນາ



Pair programming Plactice

- ❑ ຂໍ້ດີຂອງການຂຽນໂປຣແກຣມແບບເປັນຄູ່
 - ເຮົາຈະໄດ້ພັດທະນາທັກສະການເຮັດວຽກເປັນທີມ, ຝຶກການເວົ້າ, ອະທິບາຍໃຫ້ໝູ່ເຂົ້າໃຈວ່າກຳລັງຄົ້ນຄິດເລື່ອງຫຍັງ ແລະ ກຳລັງເຮັດຫຍັງຢູ່.
 - ຊ່ວຍລຸດຄວາມຜິດພາດໃນການຂຽນໂຄດໂປຣແກຣມ ເພາະວ່າມີຄົນໜຶ່ງເປັນຜູ້ເບິ່ງ, ຕິດຕາມຄວາມຖືກຕ້ອງຂອງໂຄດທີ່ຂຽນ.
 - ໄດ້ຮຽນຮູ້ເທັກນິກການຂຽນໂປຣແກຣມ ແລະ ການແກ້ໄຂປັນຫາຈາກຄູ່ ຫຼື ທີ່ມ ເຊິ່ງຈະຊ່ວຍໃຫ້ເຮົາຮຽນຮູ້ໄດ້ໄວຂຶ້ນ

Q1. ຈົ່ງຂຽນໂປຣແກຣມຮັບຄ່າຕົວເລກຈຳນວນຖ້ວນ 3 ຕົວເລກຈາກຜູ້ໃຊ້, ຖ້າຕົວເລກຫຼັກຮ້ອຍທີ່ຮັບເຂົ້າມາ ແລະ ເກັບໄວ້ໃນຕົວປ່ຽນ n ແມ່ນເລກ 3 ໃຫ້ສະແດງຄ່າ True ແຕ່ຖ້າບໍ່ແມ່ນໃຫ້ສະແດງຄ່າ False

ຕົວຢ່າງຜົນໄດ້ຮັບ

Enter a 3-digit integer: 321

True

➤ ຄໍາແນະນຳ: ເຮົາຕ້ອງຄຸ້ນເຄີຍກັບຕົວດຳເນີນການ // . ຖ້າຍັງໃຫ້ກັບໄປເບິ່ງຄືນຕື່ມອີກ.

Q2. ຈົ່ງຂຽນໂປຣແກຣມຮັບຄ່າຕົວເລກຈຳນວນຖ້ວນ, ຖ້າແມ່ນຄ່າທະວີຄູນຂອງເລກ 5 ໃຫ້ສະແດງຄ່າ True, ຖ້າບໍ່ແມ່ນໃຫ້ສະແດງຄ່າ False.

ຕົວຢ່າງຜົນໄດ້ຮັບ

Enter an integer:

125 True