



คณะวิศวกรรมศาสตร์ มหาวิทยาลัยรามคำแหง
การสอบกลางภาคการเรียนที่ 1 ปีการศึกษา 2566
วิชา CPE2222 หลักการเขียนโปรแกรม (Principle of Programming Language)

1. ลำดับจำนวนเต็ม (Integer Sequence) มีนิยามได้ดังนี้

$$K_n = K_{n-3} + 2K_{n-2} + 4K_{n-1}$$

โดยที่

$$K_2 = 3, K_1 = 2, K_0 = 1$$

จงเขียนโปรแกรมในการรับค่าตัวแปร n ที่เป็นจำนวนเต็มเท่านั้น แล้วใช้การโปรแกรมแบบเวียนซ้ำ (Recursive Programming) ในการสร้าง Tuple ที่ประกอบด้วยค่าของลำดับจำนวนเต็มนี้ตั้งแต่ K_0 จนถึง K_n ดังแสดงในตัวอย่างต่อไปนี้

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam01.py
Enter 'n' of Integer Sequence:0 (รับค่า)
A tuple of Integer Sequence [n=0] is (1,)

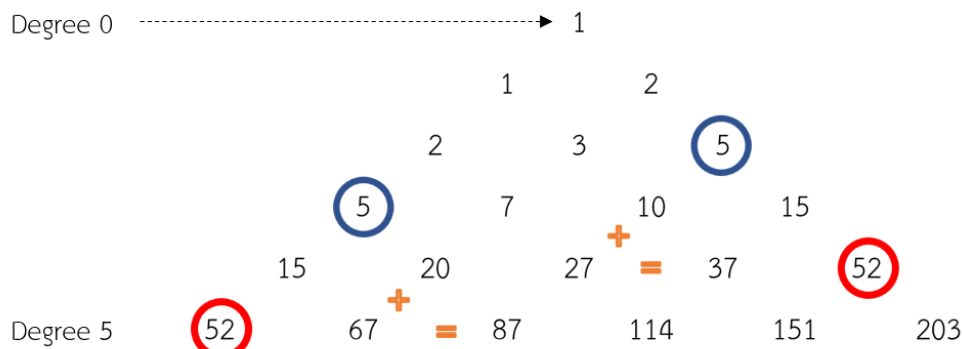
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam01.py
Enter 'n' of Integer Sequence:1 (รับค่า)
A tuple of Integer Sequence [n=1] is (1, 2)

D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam01.py
Enter 'n' of Integer Sequence:2 (รับค่า)
A tuple of Integer Sequence [n=2] is (1, 2, 3)

D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam01.py
Enter 'n' of Integer Sequence:5 (รับค่า)
A tuple of Integer Sequence [n=5] is (1, 2, 3, 17, 76, 341)

D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam01.py
Enter 'n' of Integer Sequence:10 (รับค่า)
A tuple of Integer Sequence [n=10] is (1, 2, 3, 17, 76, 341, 1533, 6890, 30967, 139181, 625548)
```

2. พิจารณาสามเหลี่ยมในรูปต่อไปนี้



โดยสามเหลี่ยมนี้แต่ละ Degree (ลำดับของแถว) จะประกอบด้วยสัมประสิทธิ์ (Coefficients) ซึ่งนำมาเขียนเป็นรายการ (List) ได้ เช่น

- Degree 0 ของสามเหลี่ยม มีรายการ (List) เป็น [1]
- Degree 1 ของสามเหลี่ยม มีรายการ (List) เป็น [1,2]
- Degree 2 ของสามเหลี่ยม มีรายการ (List) เป็น [2,3,5]
- Degree 3 ของสามเหลี่ยม มีรายการ (List) เป็น [5,7,10,15]
- Degree 4 ของสามเหลี่ยม มีรายการ (List) เป็น [15,20,27,37,52]
- Degree 5 ของสามเหลี่ยม มีรายการ (List) เป็น [52,67,87,114,151,203]... เป็นต้น

และจะเห็นได้ว่าค่าสัมประสิทธิ์สามารถถูกคำนวณได้จากค่าของสัมประสิทธิ์ของ Degree ก่อนหน้าดังแสดงในรูป จงเขียนโปรแกรมรับค่า Degree ของสามเหลี่ยมนี้ แล้วแสดงรายการ (List) สัมประสิทธิ์ของ Degree นั้น ดังแสดงตัวอย่างต่อไปนี้

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam02.py
Please Enter Degree: 0 (รับค่า)
[1]

D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam02.py
Please Enter Degree: 1 (รับค่า)
[1, 2]

D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam02.py
Please Enter Degree: 3 (รับค่า)
[5, 7, 10, 15]

D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam02.py
Please Enter Degree: 5 (รับค่า)
[52, 67, 87, 114, 151, 203]

D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam02.py
Please Enter Degree: 10 (รับค่า)
[115975, 137122, 162409, 192713, 229114, 272947, 325869, 389946, 467767, 562595, 678570]
```

3. จงเขียนโปรแกรมรับค่าความสูง (Height) ของรูปสามเหลี่ยมมุมฉาก (Right Triangle) แล้วใช้อักขระ '*' วาดรูปสามเหลี่ยมมุมฉากดังแสดงในรูปต่อไปนี้ โดยโปรแกรมจะวนซ้ำรับค่าไปเรื่อยๆ จะกว่าจะพิมพ์เลข '0'

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam03.py
----- Drawing the right triangle by "*" -----
[To quit this program by pressing "0"]
-----
Enter the height of right triangle:1 (รับค่า)
*
Enter the height of right triangle:2 (รับค่า)
*
**
Enter the height of right triangle:3 (รับค่า)
*
**
***
Enter the height of right triangle:4 (รับค่า)
*
**
* *
****
Enter the height of right triangle:5 (รับค่า)
*
**
* *
* *
*****
Enter the height of right triangle:8 (รับค่า)
*
**
* *
* *
* *
* *
* *
* *
*****
Enter the height of right triangle:0 (รับค่า)

D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>
```

4. จงเขียนโปรแกรมการทาย (Guessing) ตัวเลข 0 – 9 จำนวน 4 ตำแหน่ง โดยตัวเลขแต่ละตำแหน่งมีค่าไม่ซ้ำกัน ในการทายแต่ละครั้งโปรแกรมจะให้คำใบ้ (Hint) ว่ามีจำนวนของผู้ชาย (Men) และผู้หญิง (Women) กี่คน โดยจำนวนของผู้ชาย หมายถึง จำนวนของตัวเลขที่ทายได้ถูกต้องทั้งตัวเลขและตำแหน่ง ส่วนจำนวนของผู้หญิง หมายถึง จำนวนของตัวเลขที่ทายได้ถูกต้องเฉพาะตัวเลขเท่านั้นแต่ตำแหน่งไม่ถูกต้อง และเมื่อผู้ใช้ทายตัวเลขได้ถูกต้องแล้ว โปรแกรมจะแสดงจำนวนครั้งของการทายทั้งหมดด้วย โดยหากผู้ใช้พิมพ์ '0' จะเป็นการออกจากโปรแกรม
- ให้นักศึกษาทดสอบโปรแกรมจนกว่าจะสามารถทายผลลัพธ์ได้ถูกต้อง ดังแสดงในตัวอย่างต่อไปนี้

ตัวอย่าง: กรณีทายถูกต้อง

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam04.py
----- Welcome to MEN and WOMEN number guessing game -----
[To exit the program, press 0]
-----
Enter 4-digits number:1234 (รับค่า)
Hint: MEN = 0 and WOMEN = 1
Enter 4-digits number:5678 (รับค่า)
Hint: MEN = 0 and WOMEN = 3
Enter 4-digits number:8761 (รับค่า)
Hint: MEN = 2 and WOMEN = 1
Enter 4-digits number:8726 (รับค่า)
Hint: MEN = 3 and WOMEN = 0
Enter 4-digits number:8736 (รับค่า)
Hint: MEN = 3 and WOMEN = 0
Enter 4-digits number:8746 (รับค่า)
*** Congratuation *** your guess is correct, after 6 times
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>|
```

ตัวอย่าง: กรณีพิมพ์ '0'

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam04.py
----- Welcome to MEN and WOMEN number guessing game -----
[To exit the program, press 0]
-----
Enter 4-digits number:0 (รับค่า)
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>|
```

5. จงเขียนโปรแกรมการทอนเงินของการซื้อสินค้า โดยมีหลักการทำงานดังต่อไปนี้

- โปรแกรมจะทำการรับค่าราคาของสินค้า (Total price of products) [กำหนดให้เป็นจำนวนเต็มเท่านั้น]
- โปรแกรมจะทำการรับค่าจำนวนเงินที่ลูกค้าจ่าย (Customer payment) [กำหนดให้เป็นจำนวนเต็มเท่านั้น]
- การทอนเงินจะต้องใช้จำนวนของธนบัตรหรือเหรียญน้อยที่สุดเท่าที่จะเป็นไปได้ ประกอบด้วย
ธนบัตรใบละ 500 บาท (The 500\$Baht Banknote),
ธนบัตรใบละ 100 บาท (The 100\$Baht Banknote),
ธนบัตรใบละ 50 บาท (The 50\$Baht Banknote),
ธนบัตรใบละ 20 บาท (The 20\$Baht Banknote),
เหรียญ 10 บาท (The 10\$Baht Coin),
เหรียญ 5 บาท (The 5\$Baht Coin),
เหรียญ 2 บาท (The 2\$Baht Coin),
และ เหรียญ 1 บาท (The 1\$Baht Coin)
- โปรแกรมจะแสดงผลธนบัตรและเหรียญที่ต้องทอนแก่ลูกค้า
- กรณีที่ไม่จำเป็นต้องทอนเงิน ให้แสดงข้อความ “Complete payment”
- กรณีที่มีการกรอกข้อมูลผิดพลาดไม่สามารถทอนเงินได้ ให้แสดงข้อความ “!!! Incorrect payment !!!”

ให้นักศึกษาทดสอบการทำงานของโปรแกรกดังแสดงในรูปต่อไปนี้

กรณีที่ 1

```
-----  
The total price of products: 2141 (รับค่า)  
Customer payment: 3000 (รับค่า)  
-----  
List of money return  
The 500-$Baht Banknote = 1 Piece(s)  
The 100-$Baht Banknote = 3 Piece(s)  
The 50-$Baht Banknote = 1 Piece(s)  
The 5-$Baht Coin = 1 Coin(s)  
The 2-$Baht Coin = 2 Coin(s)  
-----
```

กรณีที่ 2

```
-----  
The total price of products: 1567 (รับค่า)  
Customer payment: 2000 (รับค่า)  
-----  
List of money return  
The 100-$Baht Banknote = 4 Piece(s)  
The 20-$Baht Banknote = 1 Piece(s)  
The 10-$Baht Coin = 1 Coin(s)  
The 2-$Baht Coin = 1 Coin(s)  
The 1-$Baht Coin = 1 Coin(s)  
-----
```

กรณีที่ 3

```
-----  
The total price of products: 1545 (รับค่า)  
Customer payment: 1544 (รับค่า)  
-----  
!!! Incorrect payment !!!  
-----
```

กรณีที่ 4

```
-----  
The total price of products: 1545 (รับค่า)  
Customer payment: 1545 (รับค่า)  
-----  
Complete payment  
-----
```

6. จงเขียนโปรแกรมรับค่าการนำข้อมูล (Plaintext) และทำการรับค่าตัวเลขเป็นกุญแจลับ (Secret Key) แล้วทำการสร้างข้อมูลที่เข้ารหัสลับ (Ciphertext) โดยการนำเอาตัวอักษรของ Plaintext มา XOR กับตัวอักษรของ Secret Key ตามตำแหน่งของตัวอักษร ดังตัวอย่างต่อไปนี้

Plaintext = "Python3" และ Secret Key = "CPE2222"

ดังนั้น Ciphertext จะคำนวณได้จากการจับคู่ XOR กันของ

'P' XOR 'C', 'y' XOR 'P', 't' XOR 'E', 'h' XOR '2', 'o' XOR '2', 'n' XOR '2', '3' XOR '2'

ในกรณีที่ Plaintext มีจำนวนตัวอักษรน้อยกว่า Secret Key ให้ตัดตัวอักษรของ Secret Key ออกให้มีขนาดเท่ากับ Plaintext

ยกตัวอย่างเช่น Plaintext = "abcde" มีจำนวน 5 ตัวอักษรและ Secret Key = "123456789" มีจำนวน 9 ตัวอักษร ดังนั้นจะได้ Secret Key = "12345" ให้เหลือจำนวนตัวอักษรเท่ากับ Plaintext (5 ตัวอักษร)

ในกรณีที่ Plaintext มีจำนวนตัวอักษรมากกว่า Secret Keyให้นำตัวอักษรของ Secret Key มาใช้ซ้ำใหม่โดยเริ่มจากตัวอักษรตัวแรกจนกระทั่งมีขนาดเท่ากับ Plaintext

ยกตัวอย่างเช่น Plaintext = "abcdefghi" จำนวน 9 ตัวอักษรและ Secret Key = "12345" จำนวน 5 ตัวอักษร ดังนั้นจะได้ Secret Key = "123451234" ให้มีตัวอักษรเท่ากับ Plaintext (9 ตัวอักษร)

ทำการทดสอบการถอดรหัสลับเพื่อให้ได้ข้อมูลเดิม (Plaintext) ตามตัวอย่างต่อไปนี้ และแสดงขนาดของ Plaintext, Secret Key และ Ciphertext ด้วย

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam06.py
Enter your plaintext:CPE2222 (รับค่า)
[The length of paintext is 7 characters]
Enter your secret key:Python3 (รับค่า)
[The length of secret key is 7 characters]
The encrypted ciphertext:[]1Z]\[]
[The length of ciphertext is 7 characters]
The decryption results:CPE2222

D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam06.py
Enter your plaintext:You got A (รับค่า)
[The length of paintext is 9 characters]
Enter your secret key:ru-eng (รับค่า)
[The length of secret key is 6 characters]
The encrypted ciphertext:+[]XE []U\
[The length of ciphertext is 9 characters]
The decryption results:You got A

D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam06.py
Enter your plaintext:Thailand (รับค่า)
[The length of paintext is 8 characters]
Enter your secret key:programming (รับค่า)
[The length of secret key is 11 characters]
The encrypted ciphertext:$[] [] []
[The length of ciphertext is 8 characters]
The decryption results:Thailand
```

หมายเหตุ: Ciphertext ในบาง Version ของ Microsoft Windows อาจจะมีการแสดงอักขระพิเศษที่แตกต่างกัน เช่น

!!)1Z]\@ → 0)1Z]\0

+→XE ♠U1 → +0XE 0U1

@!!@♀→ ♣♦♠ → \$000

7. กำหนดให้แฟ้มข้อมูลชื่อ “module_exam.py” ประกอบข้อมูลแบบพจนานุกรมซ้อนพจนานุกรม (Nested Dictionary) ชื่อ “Info_Dict” และฟังก์ชัน (Function) ชื่อ “grading”

- พจนานุกรมชื่อ “Info_Dict” มีข้อมูลคะแนนสอบ 5 วิชา (Mathematics, Physics, English, Chemistry และ Computer) ของนักศึกษาทั้งหมด 30 คน และมีโครงสร้างดังนี้
{‘66001’: {‘Mathematics’: 18, ‘Physics’: 44, ‘English’: 16, ‘Chemistry’: 30, ‘Computer’: 83},
‘66002’: {‘Mathematics’: 58, ‘Physics’: 47, ‘English’: 79, ‘Chemistry’: 90, ‘Computer’: 95},
...
‘66029’: {‘Mathematics’: 18, ‘Physics’: 81, ‘English’: 55, ‘Chemistry’: 79, ‘Computer’: 61},
‘66030’: {‘Mathematics’: 24, ‘Physics’: 49, ‘English’: 32, ‘Chemistry’: 27, ‘Computer’: 69}}

จากโครงสร้างพจนานุกรมนี้ “66001,66002,...,66029 และ 66030” เป็นรหัสประจำตัวของนักศึกษาที่ใช้เป็นกุญแจ (Key) ไปยังพจนานุกรมย่อย (Sub-dictionary) ที่ประกอบด้วยคะแนนสอบทั้ง 5 วิชาของนักศึกษาคนนั้น

- ฟังก์ชันชื่อ “grading” มีหน้าที่ในการคำนวณผลการเรียน (A,B+,B,C+,C,D+,D และ F) ของแต่ละวิชา (Mathematics, Physics, English, Chemistry และ Computer) จากคะแนนสอบของนักศึกษา โดยค่าตัวแปรเสริม (Arguments หรือ Parameters) มี 2 ตัวแปรได้แก่ ตัวแปรที่หนึ่งคือชื่อวิชา (ตัวอักษรจะต้องตรงกันกับในพจนานุกรม) และตัวแปรที่สองคือคะแนนของวิชานั้น และฟังก์ชันชื่อ “grading” จะส่งค่ากลับเป็นผลการเรียน (A,B+,B,C+,C,D+,D และ F) ดังตัวอย่างต่อไปนี้

grading(‘Physics’,18) = “F”

จงเขียนโปรแกรมรับค่าชื่อวิชา โดยหากชื่อวิชาไม่ตรงกันกับในพจนานุกรมในแฟ้มข้อมูลชื่อ “module_exam.py” ให้โปรแกรมหยุดทำงานและพิมพ์ข้อความ “!!! Subject Error !!!” ดังแสดงในตัวอย่างต่อไปนี้

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam07.py
Enter your subject [Mathematics, Physics, English, Chemistry, Computer]:computer (รับค่า)
-----
!!! Subject Error !!!
-----
```

หากพิมพ์ชื่อวิชาถูกต้อง ให้โปรแกรมแสดงจำนวนนักศึกษาที่ผลการเรียนต่างๆ (A,B+,B,C+,C,D+,D และ F) และผลคะแนนเฉลี่ยของวิชานี้ ดังแสดงในตัวอย่างต่อไปนี้


```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam07.py
Enter your subject [Mathematics, Physics, English, Chemistry, Computer]:Mathematics (รับค่า)
-----
Grade          A number of students (Percentage)
A              5 (16.67)
B+             2 ( 6.67)
B              0 ( 0.00)
C+             1 ( 3.33)
C              5 (16.67)
D+             1 ( 3.33)
D              2 ( 6.67)
F             14 (46.67)
-----
Average Score = 47.93
-----
```

และ

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam07.py
Enter your subject [Mathematics, Physics, English, Chemistry, Computer]:Computer (รับค่า)
-----
Grade          A number of students (Percentage)
A              5 (16.67)
B+             2 ( 6.67)
B              3 (10.00)
C+             3 (10.00)
C              3 (10.00)
D+             3 (10.00)
D              3 (10.00)
F              8 (26.67)
-----
Average Score = 57.73
-----
```

หมายเหตุ: กำหนดให้นักศึกษาเขียนโปรแกรมเชื่อมต่อกับแฟ้มข้อมูลชื่อ “module_exam.py” เท่านั้น ห้ามแก้ไขหรือคัดลอกข้อมูลจากแฟ้มข้อมูลชื่อ “module_exam.py”

8. จงเขียนโปรแกรมรับค่าจำนวนแถว (Rows) และจำนวนสดมภ์ (Columns) ของ Matrix “A” โดยมีการทำงานตามคำสั่งต่อไปนี้

- หากค่าของจำนวนแถว (Rows) หรือจำนวนสดมภ์ (Columns) มีค่ามากกว่า 3 ให้แสดงข้อความ “!!! Error in a number of rows or columns !!!” และหยุดการทำงานของโปรแกรม ดังแสดงในภาพต่อไปนี้

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam08.py
Enter a number of rows: 3 (รับค่า)
Enter a number of columns: 4 (รับค่า)
!!! Error in a number of rows or columns !!!
```

- ให้ทำการสุ่ม (Random) ค่าจำนวนเต็ม 0 ถึง 9 ให้กับสมาชิกทุกตัวของ Matrix A (สมาชิกแต่ละตัวอาจมีค่าซ้ำกันได้)
- คำนวณหา Matrix A^T (Transpose of Matrix “A”)
- ทำการคูณ AA^T พร้อมทั้งแสดงรายละเอียดการคำนวณ

กำหนดให้โปรแกรมแสดงผลลัพธ์ ดังแสดงในรูปต่อไปนี้

การทดสอบ $A_{1 \times 3}$

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam08.py
Enter a number of rows: 1 (รับค่า)
Enter a number of columns: 3 (รับค่า)
=====
Matrix A [1x3] with a python LIST structure : [[6, 3, 8]]
Its transpose [3x1] : [[6], [3], [8]]
=====
Calculation details for a multiplication of a matrix A and its transpose

(6x6)+(3x3)+(8x8)=109

Multiplication result [1x1] : [[109]]
=====
```

การทดสอบ $A_{2 \times 2}$

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam08.py
Enter a number of rows: 2 (รับค่า)
Enter a number of columns: 2 (รับค่า)
=====
Matrix A [2x2] with a python LIST structure : [[7, 4], [3, 1]]
Its transpose [2x2] : [[7, 3], [4, 1]]
=====
Calculation details for a multiplication of a matrix A and its transpose

(7x7)+(4x4)=65          (7x3)+(4x1)=25
(3x7)+(1x4)=25          (3x3)+(1x1)=10

Multiplication result [2x2] : [[65, 25], [25, 10]]
=====
```

การทดสอบ $A_{2 \times 3}$

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam08.py
Enter a number of rows: 2 (รับค่า)
Enter a number of columns: 3 (รับค่า)
=====
Matrix A [2x3] with a python LIST structure : [[3, 2, 7], [9, 2, 2]]
Its transpose [3x2] : [[3, 9], [2, 2], [7, 2]]
=====
Calculation details for a multiplication of a matrix A and its transpose

(3x3)+(2x2)+(7x7)=62          (3x9)+(2x2)+(7x2)=45
(9x3)+(2x2)+(2x7)=45          (9x9)+(2x2)+(2x2)=89

Multiplication result [2x2] : [[62, 45], [45, 89]]
=====
```

การทดสอบ $A_{3 \times 2}$

```
D:\Class Preparation\CPE2222\Exam\1-2566\67510010109>python exam08.py
Enter a number of rows: 3 (รับค่า)
Enter a number of columns: 2 (รับค่า)
=====
Matrix A [3x2] with a python LIST structure : [[4, 5], [3, 4], [9, 9]]
Its transpose [2x3] : [[4, 3, 9], [5, 4, 9]]
=====
Calculation details for a multiplication of a matrix A and its transpose

(4x4)+(5x5)=41          (4x3)+(5x4)=32          (4x9)+(5x9)=81
(3x4)+(4x5)=32          (3x3)+(4x4)=25          (3x9)+(4x9)=63
(9x4)+(9x5)=81          (9x3)+(9x4)=63          (9x9)+(9x9)=162

Multiplication result [3x3] : [[41, 32, 81], [32, 25, 63], [81, 63, 162]]
=====
```