

Computational Science (SC3131011) Spring 2019, Homework 1:

Due date Feb 7, 11:59 pm via google classroom

ชื่อ:

รหัสนักศึกษา:

Section:

1) (5 pts) เขียนโปรแกรม Python เพื่อคำนวณหาจำนวนเงินที่ได้จากธนาคารหลังฝากทิ้งไว้ เป็นจำนวน n ปี (exercice 1.6)

กำหนดให้ x_0 = จำนวนเงินฝากตั้งต้น

หลังจากผ่านไป n ปี

$$x_n = x_0 \left(1 + \frac{\text{อัตราดอกเบี้ย}}{100}\right)^n$$

เขียน โปรแกรม Python เพื่อคำนวณหาค่า เงินที่จะได้รับ หลังจากผ่านไป n ปี

ถ้าค่าเงินตั้งต้นคือ 1000 บาท ธนาคารให้ดอกเบี้ยร้อยละ 5 ต่อปี หลังจากสามปีผ่านไป เราจะมีเงินในธนาคารเท่าไร

In [1]:

```
# เขียนโค้ดสำหรับ โจทย์ข้อที่ 1
```

หลังจากสามปีผ่านไป เราจะมีเงินในธนาคาร =

2) (20 pts) The bell-shaped Gaussian function (modified from exercice 1.10)

สมการของ normal distribution หรือ Gaussian function สามารถเขียนให้อยู่ในรูปของ

$$f(x, m, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-m)^2}{2\sigma^2}}$$

https://en.wikipedia.org/wiki/Normal_distribution (https://en.wikipedia.org/wiki/Normal_distribution)

ให้นักศึกษาเขียนโค้ด Python เพื่อทำการ plot กราฟของ gaussian distribution โดยกำหนดให้ ค่าโดเมนของ x มีค่าตั้งแต่ -100 ถึง 100 นักศึกษาสามารถเลือกจำนวนของช่วงภายใน -100 ถึง 100 ได้ตามความเหมาะสม

2.1 ทำการ plot Gaussian distribution เมื่อ $m = 0$, $\sigma = 10$ และ $m = 0$, $\sigma = 5$ ลงบนกราฟเดียวกัน

In [2]:

```
#เขียน function f(x,m,s)

# สร้าง array ของโดเมน x

# คำนวณหาค่า f(x,m,s)

# เขียนโค้ดเพื่อทำการ Plot กราฟ และแสดงผลของกราฟ
# m = 0, s = 10
# m = 10, s = 5
```

2.2 เขียนโค้ดเพื่อทำการ integrate พื้นที่ใต้กราฟของ gaussian distribution $N(m=0, \sigma=10)$ จาก -10 ถึง 10 โดยอาศัยวิธีการประมาณพื้นที่ใต้กราฟโดยอาศัย Mid point Trapezoidal rule นักศึกษาสามารถกำหนดค่า N ได้ตามความเหมาะสม

Midpoint Trapezoidal rule (Text book หน้า 132) คือการแบ่งพื้นที่ใต้กราฟออกเป็นพื้นที่สี่เหลี่ยมผืนผ้าขนาดความกว้าง h เท่ากัน โดยให้ความสูงของรูปสี่เหลี่ยมย่อยลำดับที่ i เป็น $f(a + ih + \frac{h}{2})$ เมื่อ a คือค่าเริ่มต้นของช่วงการ integrate ดังนั้นเราสามารถประมาณค่าพื้นที่ใต้กราฟได้ดังนี้

Mid point Trapezoidal rule

$$\int_a^b f(x)dx \approx h \sum_{i=0}^{N-1} f(a + ih + \frac{h}{2})$$

ทบทวนสมการสำหรับสองวิธีที่เราเคยเห็นใน Lab และในชั้นเรียน

Trapezoidal rule

$$\int_a^b f(x)dx \approx \sum_{k=1}^N \frac{f(x_{k-1}) + f(x_k)}{2} (\delta x_k)$$

Simpson's rule (N ต้องเป็นจำนวนคู่)

$$\int_a^b f(x)dx \approx \frac{b-a}{3N} \left(f(a) + f(b) + 4 \sum_{n=1}^{N/2} f(a + (2n-1)h) + 2 \sum_{n=1}^{N/2-1} f(a + 2nh) \right)$$

$$h = \frac{b-a}{N}$$

In [3]:

```
#Implement code ของนักศึกษาที่นี่ เขียนโค้ดเพื่อทำการ integrate พื้นที่ใต้กราฟของ
#gaussian distribution N(m=0,σ=10)N(m=0,σ=10) จาก -10 ถึง 10
#โดยอาศัยวิธีการประมาณพื้นที่ใต้กราฟโดยอาศัย Mid point Trapezoidal rule
```

ผลลัพธ์ที่ได้คือ พื้นที่ =

3) (5 pts) หา Errors ใน code ที่กำหนดให้ต่อไปนี้และทำการแก้ไข ให้รันได้อย่างถูกต้อง (from exercise 1.17)

กำหนดให้ quadratic equation

$$ax^2 + bx + c = 0$$

ค่ารากของสมการสามารถหาได้จาก

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

และ

$$x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

จงอธิบายว่าปัญหาที่เกิดจาก โค้ดในการหารากของสมการ quadratic ที่กำหนดให้คืออะไร และทำการแก้ไขโค้ดให้ run ได้อย่างถูกต้อง

In [4]:

```
a = 2; b = 1; c = 2
from math import sqrt
q = b*b - 4*a*c
q_sr = sqrt(q)
x1 = (-b + q_sr)/2*a
x2 = (-b - q_sr)/2*a
print (x1, x2)
```

```
-----
-
ValueError                                Traceback (most recent call las
t)
<ipython-input-4-776c6c1997c7> in <module>()
      2 from math import sqrt
      3 q = b*b - 4*a*c
----> 4 q_sr = sqrt(q)
      5 x1 = (-b + q_sr)/2*a
      6 x2 = (-b - q_sr)/2*a
```

ValueError: math domain error

อธิบายปัญหาของโค้ดข้างต้น:

In []:

```
# ให้นักศึกษาแก้ไขโค้ดข้างต้นให้ run ได้ผลลัพธ์ที่ถูกต้อง
```

4 (10 pts) List indexing

กำหนดให้

In []:

```
q = [[['a'], 'b', 'c'], [['d', 'e', 'f']], ['g', 'h']]
```

4.1 ทำการ indexing list q เพื่อทำการเข้าถึง

- ตัวอักษร a
- list ['d', 'e', 'f']

In []:

```
# Add code เพื่อ indexing ตัวอักษร a
```

```
# Add code เพื่อ indexing list ['d', 'e', 'f']
```

In []:

```
# จากโค้ดที่กำหนดให้ จงตอบคำถามเกี่ยวกับชนิดของตัวแปร i, j
for i in q:
    for j in range(len(i)):
        print (i[j])
```

ชนิดของตัวแปร i คืออะไร

ชนิดของตัวแปร j คืออะไร

5 (10 pts) ให้นักศึกษาอ่านโค้ดของโปรแกรมข้างล่าง แล้วอธิบายว่าโค้ดแต่ละบรรทัดว่าทำงานอะไร สังเกตผลลัพธ์ที่ได้จากการรันโค้ด นักศึกษาสามารถสรุปอะไรได้บ้างจากการสังเกต

In []:

```
eps = 1.0
while (1.0 != 1.0 + eps):
    print ('.....', eps)
    eps = eps/2.0
print ('final eps:', eps)
```

อธิบายการทำงานของโค้ดข้างต้น:

ผลสรุปที่ได้จากการสังเกตการรัน Code:

6 (10 pts) Exercise 2.19: Explore round-off errors from a large number of inverse operations

บางทีนักศึกษาอาจเคยลองกดสแควร์รทบนเครื่องคิดเลขหลาย ๆ ครั้งและจากนั้นทำการยกกำลังสองด้วยจำนวนครั้งที่เท่ากัน ซึ่งค่าที่ได้ก็ควรจะเป็นค่าเริ่มต้นเหมือนเดิม แต่สิ่งนี้ไม่ได้เกิดขึ้นเสมอไป

1) ให้นักศึกษาอธิบายการทำงานของโค้ดโปรแกรมข้างล่าง

2) อธิบายว่าเพราะเหตุใด ผลลัพธ์สุดท้าย ค่า r จึงไม่ได้ 2

In []:

```
from math import sqrt
T = 60
for n in range(1, T):
    r = 2.0
    for i in range(n):
        r = sqrt(r)
    for i in range(n):
        r = r**2
    print ("%d times sqrt and **2: %.16f" % (n, r))
```