



คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

การสอบภาคปฏิบัติ ครั้งที่ 1 ภาคเรียนที่ 1 ปีการศึกษา 2562

รหัสวิชา 01076005, 01076249 Data Structures and Algorithm ชั้นปี 2 ห้อง D และนักศึกษาลงทะเบียน

สอบ วันอังคาร ที่ 1 ตุลาคม พ.ศ. 2562 เวลา 180 นาที

---

คำเตือน นักศึกษาซึ่งทุจริตในการสอบ จะไม่ได้รับการพิจารณาผลการเรียนในภาคการศึกษาที่นักศึกษา  
กระทำการทุจริตนั้น และพักการเรียนในภาคการศึกษาปกติถัดไปอีก 1 ภาคการศึกษา

- คำสั่ง
1. ข้อสอบมี 3 ข้อ ข้อละ 5 คะแนน
  2. คิดร้อยละ 15 ของคะแนนรวมทั้งวิชา
  3. ห้ามนำอุปกรณ์สื่อสารติดตัวเข้าห้องสอบ
  4. การคิดคะแนนแต่ละข้อไม่มีการแบ่งส่วนการให้คะแนน
  5. ห้ามนำอาหารและเครื่องดื่มเข้าห้องสอบ
  6. นักศึกษาช่วงเข้าห้ามออกจากห้องสอบก่อน 12:30 สามารถเตรียมหนังสืออ่านเล่นติดตามได้

ชื่อ - นามสกุล ..... รหัสนักศึกษา.....

## 1. จงเขียน class โดยมี method ดังต่อไปนี้

```
class Stack :
    def __init__(self, list) :    #สร้าง empty list
    def __str__(self) :          #พิมพ์ของที่อยู่ใน stack จากล่างขึ้นบน
    def __len__(self) :          #ส่งค่าจำนวนสมาชิกใน stack
    def isEmpty(self) :          #ส่งค่า True ถ้า Stack ว่าง, False ถ้าไม่ว่าง
    def push(self, i) :          #นำของเก็บใน stack
    def pop(self) :              #นำของออกจาก stack
```

โดย method ที่เขียนจะต้องทำงานได้อย่างถูกต้องแล้วนำมาใช้เขียนโปรแกรม infix to postfix และแสดงจำนวน operation และ operand

ตัวอย่างการทำงาน

```
Enter infix expression      : a+b*c-d
Result postfix expression   : abc*+d-
Number of operation         : 3
Number of operand          : 4
```

```
Enter infix expression      : a+b*c-(d/e+f)*g
Result postfix expression   : abc*+de/f+g*-
Number of operation         : 6
Number of operand          : 7
```

```
Enter infix expression      : (a+b)*c/d
Result postfix expression   : ab+c*d/
Number of operation         : 3
Number of operand          : 4
```

2. จงสร้าง class LinkedList ซึ่งมี method ดังต่อไปนี้เพื่อใช้เขียนโปรแกรม

```
class LinkedList :
    class Node :
        def __init__(self,data,next) :    #สร้างโหนดเพื่อเก็บข้อมูล
        def __init__(self):                #สร้าง linked list ว่าง
        def __str__(self) :                #แสดงข้อมูลจากหัวไปหาง
        def __len__(self) :                #ส่งค่าจำนวนสมาชิก
        def isEmpty(self) :                #ส่งค่า True ถ้า linked list ว่าง, False ถ้าไม่ว่าง
        def append(self,data) :            #เพิ่มข้อมูลต่อท้าย
        def remove(self,data) :            #ลบข้อมูลตัวนั้นออก
        def add(self,data) :                #ตามข้อกำหนดข้อ 2.1
```

ให้นักศึกษา เขียน method ที่ทำงานได้อย่างถูกต้อง ของ class LinkedList ให้สมบูรณ์

2.1 def add(self,data) เป็น method ที่จะเพิ่มข้อมูลโดยการเรียงลำดับข้อมูลที่เพิ่มเข้าไป ดังตัวอย่างการทำงานด้านล่าง

ตัวอย่างการทำงาน

```
L1 = LinkedList()
L1.add(5)           #ข้อมูลใน L1  5
L1.add(9)           #ข้อมูลใน L1  5 -> 9
L1.add(7)           #ข้อมูลใน L1  5 -> 7 -> 9
L1.add(3)           #ข้อมูลใน L1  3 -> 5 -> 7 -> 9
L1.add(11)          #ข้อมูลใน L1  3 -> 5 -> 7 -> 9 -> 11
```

2. ให้นักศึกษาใช้ class LinkedList ข้างต้นเขียนโปรแกรมเพื่อหาค่าเฉลี่ยเลขคณิต (mean) ค่าฐานนิยม (mode) และ ค่ามัธยฐาน (median)

ตัวอย่างการทำงาน

```
Enter 12 number : 7 9 3 2 1 2 3 4 8 9 3 15
Output :
LinkedList data : 1 2 2 3 3 3 4 7 8 9 9 15
Mean = 5.5
Mode = 3
Median = 3.5
```

```
Enter 12 number : 7 6 74 44 6 37 55 35 3 31 3 10
Output :
LinkedList data : 3 3 6 6 7 10 31 35 37 44 55 74
Mean = 25.92
Mode = 3, 6
Median = 20.5
```

3. จงเขียนโปรแกรมเพื่อช่วยจำลองการจัดคิวการประมวลผลงานในบริษัทแห่งหนึ่งโดยมีเงื่อนไขดังนี้

1. ผู้ใช้ระบบ หรือ พนักงานในบริษัทมีสิทธิได้รับการเร็วหรือช้าในการประมวลผลงานที่ส่งเข้ามารอในคิว ขึ้นอยู่กับรหัส (ID) ของพนักงานท่านนั้น

รหัส	100-199	มีสิทธิสูงสุด (งานที่ส่งจะได้รับการทำก่อนกลุ่มอื่น)
	200-299	มีสิทธิรองสูงสุด
	300-399	...
	...	...
	800-899	มีสิทธิก่อนกลุ่มต่ำสุด
	900-999	มีสิทธิต่ำสุด (งานที่ไม่เร่ง เช่น การประมวลผลงานทำช่วงวันหยุดเท่านั้น)

2. พนักงานที่มีรหัสในกลุ่มเดียวกัน งานจะทำเรียงลำดับก่อน-หลัง ตามคิวที่ส่ง ใครส่งก่อนจะได้รับบริการก่อน

3. ถ้ามีงานของพนักงานที่มีสิทธิสูงสุดอยู่ในคิว งานของเขาจะต้องได้รับการประมวลผลก่อนเสมอ แม้ว่าจะส่งงานเข้าประมวลผลที่หลังกลุ่มอื่น

ลักษณะของ Input :

A	764	Invoice
A	982	PO
A	181	Payroll
A	456	Customer
A	121	Tax
A	181	OT
L		
R		
C	982	PO
U	764	Invoice
R		

คำอธิบาย

ฟิลด์ที่ 1 คือ รหัสรายการ (code)

ฟิลด์ที่ 2 คือ รหัสพนักงาน (ID)

ฟิลด์ที่ 3 คือ ชื่อของงาน (job)

รหัสรายการมี (code) มี 6 ประเภท คือ

การสร้าง class Queue สามารถใช้โครงสร้างข้อมูล list หรือ linked list สำหรับนำมาสร้างได้ โดยต้องมี method enqueue สำหรับนำข้อมูลเข้าที่ท้ายคิว และ dequeue สำหรับนำข้อมูลออกจากหัวคิวและลบข้อมูลรายการนั้น

โดยเขียนโปรแกรมที่รับคำสั่งตามลักษณะของ input แล้วแสดงผลการทำงานดังต่อไปนี้

### 3.1 code 'A' (Add)

ให้พนักงานของพนักงานที่ส่งประมวลผล เข้ารอในคิว โดยเขียน method ชื่อ addJob(ID, jobName) โดยรับรหัสพนักงาน และ ชื่อของงาน ภายหลังจากส่งงานเข้าคิวแล้วให้แจ้งให้ผู้ใช้ทราบว่างานได้บันทึกในคิวแล้ว

ตัวอย่างการทำงาน

A      764      Invoice

Job submitted.

### 3.2 code 'L' (List)

ให้แจกแจงทุก ๆ งานที่รอในคิว ตามลำดับการทำงาน เพื่อต้องการทราบถึงปริมาณงานที่ต้องการประมวลผลอยู่ในขณะนั้นว่ามีมากน้อยเพียงใด

ตัวอย่างการทำงาน

L

ID	Job Name
181	Payroll
121	Tax
181	OT
456	Customer
764	Invoice
982	PO

### 3.3 code 'R' (Run)

เมื่อคอมพิวเตอร์ทำงานของพนักงานคนใดคนหนึ่งเสร็จเรียบร้อยแล้ว เครื่องจะเริ่มรับงานถัดไป (งานที่กำลังรอในคิว และมีสิทธิสูงสุด ณ ขณะนั้น) โดยเขียน method ชื่อ getNextJob() เพื่อแสดงรหัส (ID) และงาน (jobName) ที่จะประมวลผล งานใดที่ถูกเลือก (ตามเงื่อนไข) ให้แจ้งให้ทราบ และนำรายการนั้นออกจากคิว

ตัวอย่างการทำงาน

R

ID : 181          Job : Payroll starts running.

### 3.4 code 'C' (cancel)

เป็นไปได้ที่ในบางครั้งพนักงานคนใดคนหนึ่งได้ส่งงานเข้ามาเพื่อประมวลผลและในขณะนั้นกำลังรออยู่ในคิว ปรากฏว่ามีปัญหาบางประการ จึงเกิดเปลี่ยนใจขอยกเลิกงานที่รอในคิว หลังจากยกเลิกแล้วให้แจ้งให้ผู้ใช้ทราบ โดยเขียน method ชื่อ `cancelJob(ID,jobName)`

ตัวอย่างการทำงาน

C          982          PO

ID : 982          Job: PO has been cancelled.

### 3.5 code 'U' (Urgent)

กรณีที่พนักงานคนใดคนหนึ่งมีรหัสซึ่งมีสิทธิต่ำมาก ๆ เป็นไปได้ที่จะต้องทำงานเร่งด่วนกะทันหัน หัวหน้างานมีสิทธิดำเนินการเพื่อลัดคิวของพนักงานคนนั้นให้มาอยู่ต้นคิว โดยเขียน method ชื่อ `urgentJob(ID,jobName)`

ตัวอย่างการทำงาน

U          764          Invoice

ID : 764          Job: Invoice has first priority.