

Practice Exercise: Queue

5 กุมภาพันธ์ 2568

เรียงลำดับคำขอ (Request Handle)

เรามี Request เข้ามาเรื่อย ๆ โดยไม่รู้จบ ซึ่งแต่ละ Request จะใช้เวลาในการทำงานจำนวน x ms และแต่ละ Request จะเข้ามาที่เวลา t_i ตัวอย่างเช่น

- $x = 7$ หมายความว่าทุก Request จะใช้เวลา 7 ms เท่า ๆ กัน
- $t_1 = 0$ หมายความว่า มี Request เข้ามาที่เวลา 0 ms ดังนั้น ที่เวลา 0 จะมีจำนวน Request เท่ากับ 1 Request
 - Request นี้ จะอยู่ในการทำงานตั้งแต่ช่วงเวลาเริ่ม ms ที่ 0 ไปจนถึงจบ ms ที่ 6
 - เมื่อจบช่วงเวลา ms ที่ 6: Request นี้จะเสร็จการทำงานและออกจากโปรแกรมไป
 - เมื่อเริ่ม ms ที่ 7 จะไม่มี Request ดังกล่าวนี้อยู่ในคิวแล้ว
- $t_2 = 7$ หมายความว่า มี Request เข้ามาที่เวลา 7 ms ดังนั้น ที่เวลา ms = 7 จะมีจำนวน Request เท่ากับ 1 Request (จากเหตุผลในข้อด้านบน)
- $t_2 = 9$ หมายความว่า มี Request เข้ามาที่เวลา 9 ms ดังนั้น ที่เวลา ms = 9 จะมีจำนวน Request เท่ากับ 2 Request
- $t_3 = 13$ หมายความว่า มี Request เข้ามาที่เวลา 13 ms ดังนั้น ที่เวลา ms = 13 จะมีจำนวน Request เท่ากับ 3 Request
- $t_4 = 14$ หมายความว่า มี Request เข้ามาที่เวลา 14 ms ดังนั้น ที่เวลา ms = 14 จะมีจำนวน Request เท่ากับ 3 Request
- $t_5 = 21$ หมายความว่า มี Request เข้ามาที่เวลา 21 ms ดังนั้น ที่เวลา ms = 21 จะมีจำนวน Request เท่ากับ 1 Request

เราจะรับค่าไปเรื่อย ๆ จนกว่ารับค่า -1 เป็นรหัสของการจบโปรแกรม

ให้นักศึกษาใช้โครงสร้างข้อมูลแบบ Queue เพื่อเก็บข้อมูลของลำดับเวลาที่ Request เข้ามา และทุกครั้งที่ มี Request ตัวใหม่เพิ่มเข้ามา ให้นักศึกษาพิมพ์จำนวน Request ที่กำลังทำงานอยู่

ข้อมูลนำเข้า (Input)

บรรทัดที่ 1	จำนวนเต็ม x บอกเวลาทำงานของแต่ละ Request โดยที่ $1 \leq x \leq 100$
บรรทัดที่ 2 เป็นต้นไป	จำนวนเต็ม t_i บอกเวลาที่เข้ามาของแต่ละ Request โดยที่ $t_i < t_j$ เสมอ เมื่อ $i < j$
บรรทัดสุดท้าย	ตัวเลข -1 เพื่อจบการรับค่า

ข้อมูลส่งออก (Output)

บรรทัดที่ 1 เป็นต้นไป	จำนวนของ Request ที่กำลังทำงานอยู่
-----------------------	------------------------------------

ตัวอย่างข้อมูลนำเข้า ส่งออก (Examples of Input & Output)

Input	Output
7	1
0	1
7	2
9	3
13	3
14	1
21	
-1	
10	1
1	2
2	3
3	4
4	5
8	6
9	7
10	7
11	1
30	
-1	

แจกของ (Giveaway)

ในวันที่ 10 กุมภาพันธ์ 2568 มีกิจกรรม CPE Games 2025 ซึ่งช่วงคำของกิจกรรมก็จะมีปาร์ตี้กินเลี้ยง, แสงวงตรี, สอยดาว และกิจกรรมต่าง ๆ อีกมากมาย และกิจกรรมพิเศษสำหรับน้อง ๆ CPE38 ก็คือการแจกของขวัญจากอาจารย์ผู้สอนวิชา CPE112 นั่นเอง

มีนักศึกษาต่อแถวในคิวจำนวน n คน และมีของให้จำนวน n ชิ้น ซึ่งแต่ละชิ้นเป็นไปได้อยู่สองอย่างคือ

- 1 คือของขวัญสุดพิเศษจากอาจารย์เตย
- 2 คือของขวัญสุดพิเศษจากอาจารย์วี

โดยของจะวางเรียงจากหน้าไปหลัง โดยคนที่อยู่ลำดับหน้าสุดจะได้เห็นของที่วางอยู่หน้าสุดแต่เพียงอันเดียวเท่านั้น นักศึกษาแต่ละคนมีของอยู่ในใจที่ต้องการ (เป็นไปได้สองแบบคือ 1 หรือ 2) โดยให้คนที่อยู่แถวหน้าสุดมีสิทธิได้เห็นของก่อน โดยมีเงื่อนไขดังต่อไปนี้

- หากนักศึกษายากได้ของที่ตรงกับของที่อยู่ตรงหน้าสุด นักศึกษาจะหยิบของชิ้นนั้นไป แล้วออกไปจากคิว
- หากของที่อยู่ตรงหน้าสุด ไม่ตรงกับสิ่งที่นักศึกษายากได้ นักศึกษาจะไม่หยิบของชิ้นนั้น แล้วออกจากคิวเพื่อไปต่อแถวใหม่ด้านหลังสุด

เรามาดูตัวอย่างเหตุการณ์สมมติกัน

- ให้มีจำนวนนักศึกษาและจำนวนของขวัญ $n = 6$ ชิ้น
- ให้คิวของนักศึกษาเป็น {2, 2, 1, 2, 1, 2} และของขวัญคือ {2, 1, 1, 2, 2, 1} (ให้ตัวอักษรสีแดงคือลำดับหน้าสุดของคิว)
- นักศึกษาคนแรกได้ของที่อยากได้ ทำให้เขาหยิบของแล้วออกไปจากคิว คิวของนักศึกษากลายเป็น {2, 1, 2, 1, 2} และของขวัญเป็น {1, 1, 2, 2, 1}
- นักศึกษาคนต่อมาไม่ได้ของที่อยากได้ ทำให้เขาออกจากคิวแล้วไปต่อแถวหลังสุด คิวของนักศึกษากลายเป็น {1, 2, 1, 2, 2} และของขวัญเป็น {1, 1, 2, 2, 1}
- นักศึกษาคนต่อมาได้ของที่อยากได้ ทำให้เขาหยิบของแล้วออกไปจากคิว คิวของนักศึกษากลายเป็น {2, 1, 2, 2} และของขวัญเป็น {1, 2, 2, 1}
- นักศึกษาคนต่อมาไม่ได้ของที่อยากได้ ทำให้เขาออกจากคิวแล้วไปต่อแถวหลังสุด คิวของนักศึกษากลายเป็น {1, 2, 2, 2} และของขวัญเป็น {1, 2, 2, 1}
- ทำเช่นนี้เรื่อย ๆ กับนักศึกษาทั้งสามคนด้านหน้านี้ ทั้ง 3 คนหยิบของแล้วออกไปจากคิว จนทำให้คิวของนักศึกษาเหลือเพียง {2} และของขวัญเป็น {1}
- เราจะได้ข้อสรุปว่า จะมีนักศึกษาจำนวน 1 คนที่จะไม่ได้ของขวัญไป

ให้นักศึกษาเขียนโปรแกรมภาษาซี เพื่อหาว่าจะมีนักศึกษากี่คนที่จะได้ของขวัญ

ข้อมูลนำเข้า (Input)

บรรทัดที่ 1	จำนวนเต็ม n แทนจำนวนนักศึกษาและของขวัญ โดยที่ $1 \leq n \leq 10,000$
บรรทัดที่ 2	จำนวนเต็ม n จำนวนแทนของที่นักศึกษายากได้ โดยมีแค่ 2 ค่าที่เป็นไปได้คือ 1 หรือ 2
บรรทัดที่ 3	จำนวนเต็ม n จำนวนแทนของขวัญที่วางอยู่ โดยมีแค่ 2 ค่าที่เป็นไปได้คือ 1 หรือ 2

ข้อมูลส่งออก (Output)

บรรทัดที่ 1	จำนวนนักศึกษาที่ไม่ได้รับของขวัญ
-------------	----------------------------------

ตัวอย่างข้อมูลนำเข้า ส่งออก (Examples of Input & Output)

Input	Output
6 2 2 1 2 1 2 2 1 1 2 2 1	1
6 2 2 2 1 1 2 2 1 1 1 2 2	3
4 2 2 1 1 1 2 1 2	0

จัดการหน่วยความจำ (Memory Management)

ในระบบคอมพิวเตอร์ RAM เป็นหน่วยความจำที่จัดการกับ Process ที่ทำงานในระบบ ซึ่งควรจะมีให้เพียงพอต่อการใช้งาน

หากว่าเรามี Process มากมายที่จะทำงานซึ่งมากเกินไปกว่าหน่วยความจำของ RAM โปรแกรมเหล่านี้จะต่อแถวอยู่ใน Waiting Queue เพื่อให้ Process ที่มาก่อนสามารถเข้าไปทำงานได้ก่อน แล้วจึงสามารถเข้าไปทำงานได้เมื่อมีพื้นที่ว่าง (ที่จริงแล้วเรายังมีหน่วยความจำอื่นที่สามารถถูกเอามาใช้ให้คล้าย ๆ กับ RAM และยังมีเทคนิคการสลับ Process ทำงานด้วย แต่เรื่องพวกนั้นเอาไว้ก่อน ๆ เรียนตอนปี 3 แล้วกันนะ)

ยกตัวอย่างสถานการณ์เช่น ในคอมพิวเตอร์เรามีหน่วยความจำ RAM จำนวน 16 GB เรามี Process ที่รอคิวดังนี้ {12, 3, 2, 8, 6, 1} ให้เลขในคิวนิยามถึงพื้นที่ที่ Process นั้น ๆ ต้องการในหน่วย GB และให้สมมติว่าทุก Process ใช้เวลาทำงานเท่ากัน ในที่นี้ให้ $t = 4$ แทนเวลาที่แต่ละ Process ต้องการทำงานในหน่วยวินาที เราจะได้ผลดังนี้

- Process {12, 3} จะได้เข้าไปทำงานพร้อมกัน เพราะจะใช้พื้นที่รวมเป็น 15 GB (ไม่เกิน 16 GB) ซึ่งใช้เวลา 4 วินาทีในการทำงาน
- เมื่อเวลาผ่านไป 4 วินาที ทั้งสอง Process ด้านบนทำงานเสร็จ จะคืนพื้นที่ว่างให้ ทำให้ Process {2, 8, 6} ได้เข้าไปทำงาน ใช้เวลา 4 วินาที
- หลังจากนั้นก็จะคืนพื้นที่ให้ ให้ Process {1} ได้เข้าไปทำงาน ใช้เวลาอีก 4 วินาที จึงจะจบการทำงานทั้งหมด
- สังเกตได้ว่ากระบวนการทั้งหมดใช้เวลา 12 วินาทีจึงเสร็จสิ้น

จากตัวอย่างด้านบน หากว่าเรากำหนดให้มีเวลาทำงานทั้งหมด 12 วินาที คอมพิวเตอร์สามารถประมวลผลได้ครบทั้ง 6 Process

แต่หากว่าเรากำหนดให้คอมพิวเตอร์ทำงาน 10 วินาที Process {1} จะไม่สามารถทำงานเสร็จสิ้นได้ ทำให้จำนวน Process ที่ทำงานเสร็จมีเพียง 5 Process นั้นเอง

เมื่อเราให้พื้นที่แรมมา r GB ให้เวลาในการทำงานของแต่ละ Process เป็น t วินาที และให้คอมพิวเตอร์ทำงานเป็นเวลา n วินาที ให้นักศึกษาเขียนโปรแกรมหาว่าคอมพิวเตอร์สามารถประมวลผล Process ได้สำเร็จจำนวนกี่ Process

ข้อมูลนำเข้า (Input)

บรรทัดที่ 1	จำนวนเต็ม r , t และ n แทนพื้นที่แรม, เวลาของแต่ละ Process และเวลาที่ให้คอมพิวเตอร์ทำงาน ตามลำดับ โดยที่ $t < n$ เสมอ
บรรทัดที่ 2 เป็นต้นไป	จำนวนเต็มบวก แทนขนาดพื้นที่ที่แต่ละ Process ต้องการในหน่วย GB
บรรทัดสุดท้าย	ค่า -1 แทนการจบการรับค่า

ข้อมูลส่งออก (Output)

บรรทัดที่ 1	จำนวน Process ที่ทำงานได้สำเร็จ
-------------	---------------------------------

ตัวอย่างข้อมูลนำเข้า ส่งออก (Examples of Input & Output)

Input	Output
16 4 10 12 3 2 8 6 1 -1	5
8 3 11 6 2 10 1 4 2 -1	2

หมายเหตุ: ในตัวอย่างที่สอง พบว่ามี Process ที่มีขนาดใหญ่กว่า RAM ทำให้คิวที่เหลือดัดขัดจนหมดเวลา ทำให้มีแค่ 2 Process แรกที่รันได้สำเร็จ

อูยองอู (Extraordinary Attorney Woo)

”ชื่อของฉัน ไม่ว่าจะอ่านตามตรง หรืออ่านกลับด้านก็เป็นอูยองอูค่ะ กนก บวบ นลิน ยาย วาดดาว อูยองอู”

วันนี้เราจะมาตรวจสอบกันว่าคำที่ให้ไปนี้ อ่านกลับหลังไปหน้าก็ยังเป็นเหมือนเดิมหรือเปล่า สิ่งนี้เรียกว่า พาลินโดรม (Palindrome) คือสายอักขระที่ไม่ว่าจะสะกดไปข้างหน้าหรือย้อนกลับก็จะได้ดังเดิม

เช่นคำว่า **NOON** ก็นับเป็นพาลินโดรม เพราะสะกดจากหน้าไปหลังหรือหลังไปหน้า ก็ได้ **NOON** เหมือนกัน

ง่าย ๆ เลย ให้นักศึกษานำหลักการของ Deque (Double-ended queue) มาตรวจสอบว่า String ที่ให้มานี้เป็น Palindrome หรือไม่

ข้อมูลนำเข้า (Input)

บรรทัดที่ 1	String ที่ประกอบไปด้วยตัวอักษรอังกฤษพิมพ์ใหญ่เท่านั้น ไม่จำกัดความยาว ไม่มี Whitespace
-------------	--

ข้อมูลส่งออก (Output)

บรรทัดที่ 1	หาก String ที่ใส่มาเป็นพาลินโดรมให้พิมพ์ YES หากไม่เป็นให้พิมพ์ NO
-------------	--

ตัวอย่างข้อมูลนำเข้า ส่งออก (Examples of Input & Output)

Input	Output
NOON	YES
CPETHREEEIGHTYOUSOCUTE	NO

รหัสลับ (Secret code for you & me)

เราจะมาแปลงข้อความให้เป็นรหัสลับที่เราสองคนรู้จักกันเท่านั้น ... ด้วยการใช้ Priority Queue นั่นเองงง

โดยมีกฎการแปลงแบบแสนจะง่ายว่า:

- พยัญชนะภาษาอังกฤษทั้งหมดจะถูกให้ค่า Priority เป็น 1
- สระภาษาอังกฤษทั้งหมดจะถูกให้ค่า Priority เป็น 3
- ตัวอักษรที่เหลือที่ไม่ใช่ตัวอักษรภาษาอังกฤษ (เช่น สัญลักษณ์ หรือตัวเลข) จะถูกให้ค่า Priority เป็น 2
- ค่า Priority ที่มากที่สุดคือมีความสำคัญสูงสุด รองลงมาเรื่อย ๆ
- สำหรับตัวอักษรที่มีค่า Priority เท่ากัน ให้ใช้หลักการของ "First come, first serve"

เช่นคำว่า **I loVe You!** เราจะได้ค่า Priority ของแต่ละตัวอักษรว่า

ตัวอักษร	I	'whitespace'	l	o	V	e	'whitespace'	Y	o	u	!
ค่า Priority	3	2	1	3	1	3	2	1	3	3	2

ผลลัพธ์ที่ได้คือ **Ioeou !lVY**

ให้นักศึกษาแปลง String ที่รับเข้ามาให้เข้าตามเงื่อนไขด้านบน

ข้อมูลนำเข้า (Input)

บรรทัดที่ 1	String ที่เราต้องการแปลง
-------------	--------------------------

ข้อมูลส่งออก (Output)

บรรทัดที่ 1	String ที่ถูกแปลงตามเงื่อนไขด้านบน
-------------	------------------------------------

ตัวอย่างข้อมูลนำเข้า ส่งออก (Examples of Input & Output)

Input	Output
I loVe You!	Ioeou !lVY
error	eorrr