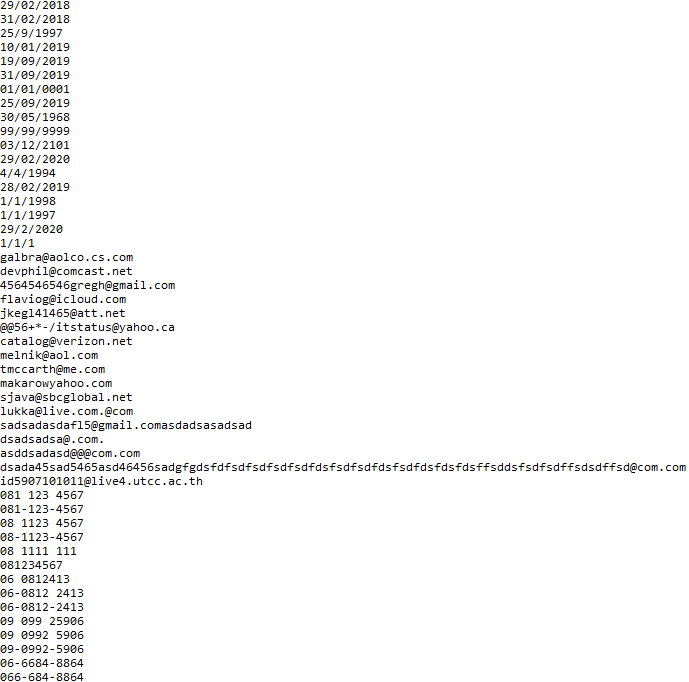
5907101011 นาย สิรวิชย์ เตชะวณิชย์

**NC-358 Assignment 3**

**จงเขียน Regular expression 3 ประโยคเพื่อ grep/egrep เพื่อกรองข้อมูลและเงื่อนไขต่อไปนี้**

****

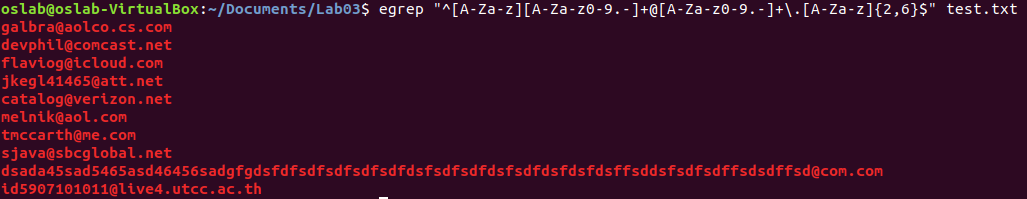
*(ตัวอย่างข้อมูลเพื่อใช้ในการกรองด้วยคำสั่ง grep/egrep)*

**1.Email เงื่อนไขคือ**

**ตัวอักษรตัวแรกของชื่ออีเมล์ขึ้นต้นด้วยตัวอักษร ตัวถัดไปเป็นอะไรก็ได้**

**มีการตรวจสอบสัญลักษณ์ @**

**ชื่อโดเมนที่อยู่หลังสัญลักษณ์ @ มีความยาวอย่างน้อย 1 ตัวอักษร ไม่ลงท้ายด้วย . (จุด) หรืออักขระพิเศษ**

****

คำตอบ

egrep "^[A-Za-z][A-Za-z0-9.-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,6}$" test.txt

**^[A-Za-z][A-Za-z0-9.-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,6}$"**

^[A-Za-z] หมายถึง ขึ้นต้นด้วยตัวอักษรพิมพ์เล็กหรือใหญ่ตามด้วยตัวอักขระอื่น ๆอะไรก็ได้ตามด้วย

[A-Za-z0-9.-] อักขระอื่น ๆ +@ จำเป็นต้องมี @ อย่างน้อย 1 ตัวอักษร [A-Za-z0-9.-] ตามด้วยอักขระอื่น ๆ

ขั้นด้วย “.” ซึ่งจำเป็นต้องมีการใช้เครื่องหมาย \ เพื่อ escape ไม่ให้เป็น regEx Dot “.” แทน และ[A-Za-z]{2,6}$" ลงท้ายด้วยอักษรตัวพิพม์เล็กหรือใหญ่อย่างน้อย 2 ตัว สูงสุดไม่เกิน 6 ตัว

**2.Mobile Number สำหรับเบอร์มือถือของเมืองไทย เงื่อนไขคือ**

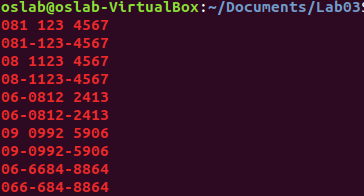
**ขึ้นต้นด้วย 06, 08, หรือ 09**

**รูปแบบจะต้องอยู่ใน 4 รูปแบบต่อไปนี้ เช่น 081 123 4567, 081-123-4567, 08 1123 4567, 08-1123-4567**

แบ่ง Group ในการค้าหาออกเป็น 2 Pattern หลัก ๆ

ซึงจะสักเกตุได้ว่า ในแต่ละ Pattern นั้นสามารถเป็นไปได้สองรูปแบบคือส่วนที่ขั้นตัวเลขนั้นสามารถเป็น “–“ หรือ “ “ ซึ่งถ้าเราเขียนแบบง่ายๆโดยใช้ [ -] นั้นสามารถผิดพลาดได้เราจึงควรวิเคราะห์คำสั่งใหม่เป็น egrep เนื่องจากเราต้องการผลลัพธ์ที่มีหลาย Pattern มากๆมีความกว้างของช่วงข้อมูลที่หลากหลาย

และสามารถขั้นแต่ละ Pattern ด้วย |



*(ตัวอย่างการใช้คำสั่ง grep แบบง่าย)*

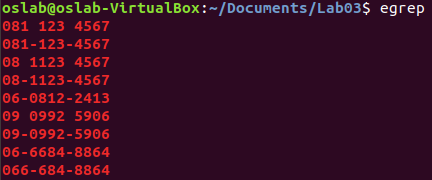
คำตอบ

*egrep "((^0[689][0-9]-[0-9]{3}-[0-9]{4})|(^0[689][0-9] [0-9]{3} [0-9]{4}))|((^0[689]-[0-9]{4})-[0-9]{4}|(^0[689] [0-9]{4} [0-9]{4}))" test.txt*

**^0[689][0-9]-[0-9]{3}-[0-9]{4}** หมายถึง ต้องการให้มีตัวอักษรแรกเป็น 0 ตามด้วย 6 , 8 หรือ 9 ตัวใดตัวหนึ่ง 1 ตัว ต้องการตัวเลข 0-9 ตัวใดก็ได้ 1 ตัว ขั้นด้วยเครื่องหมาย - ตามด้วยต้องการตัวเลข 0-9 ตัวใดก็ได้ 3 ตัว ขั้นด้วยเครื่องหมาย - ตามด้วยต้องการตัวเลข 0-9 ตัวใดก็ได้ 4 ตัว

**^0+[689]-[0-9]{4}-[0-9]{4}** หมายถึง ต้องการให้มีตัวอักษรแรกเป็น 0 ตามด้วย 6 , 8 หรือ 9 ตัวใดตัวหนึ่ง 1 ตัว ขั้นด้วยเครื่องหมาย - ตามด้วยต้องการตัวเลข 0-9 ตัวใดก็ได้ 4 ตัว ขั้นด้วยเครื่องหมาย - ตามด้วยต้องการตัวเลข 0-9 ตัวใดก็ได้ 4 ตัว

*\*\* Hint Pattern ย่อยๆที่มีความคล้ายคลึง 000-000-0000 คือ 000 000 0000 เราสามารถเปลี่ยนจาก “–“ เป็น “ “ ได้และครอบด้วย () เพื่อแบ่ง group ก่อนที่จะขั้นด้วย เครื่องหมายหรือ “|” \*\**

****

*(ผลลัพธ์จากการกรองเบอร์โทรศัพท์)*

**3.Date สำหรับกรองข้อมูลวันที่ เงื่อนไขเช่น**

**DD/MM/YYYY โดยมีเช็ควันที่ที่ถูกต้องของแต่ละเดือน เช่น เดือนก.พ. 2019 มีจำนวนวันเพียง 28 วัน เป็นต้น**

**มีการเช็คตัวเลขของเดือนที่ถูกต้อง เช่น อยู่ระหว่างเดือน 1 - 12 หรือเป็น 01 - 12 เป็นต้น**

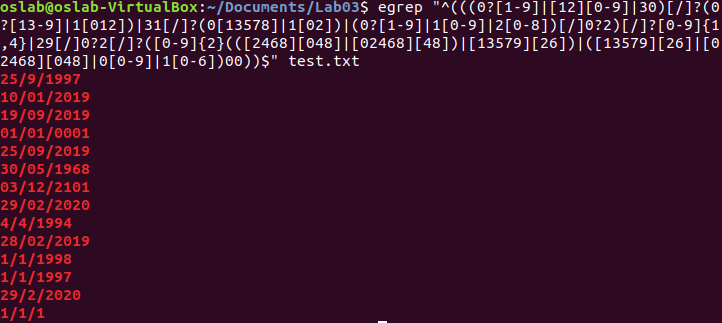
คำตอบ

**egrep "^(((0?[1-9]|[12][0-9]|30)[/]?(0?[13-9]|1[012])|31[/]?(0[13578]|1[02])|(0?[1-9]|1[0-9]|2[0-8])[/]0?2)[/]?[0-9]{1,4}|29[/]0?2[/]?([0-9]{2}(([2468][048]|[02468][48])|[13579][26])|([13579][26]|[02468][048]|0[0-9]|1[0-6])00))$" test.txt**

**"^(((0?[1-9]|[12][0-9]|30)[/]?(0?[13-9]|1[012])|31[/]?(0[13578]|1[02])| เป็นการตรวจสอบจำนวนวันกับเดือนเช่น 13578 1[02] หรือ 12 นั้นมี 31 วัน ส่วนที่ไม่เข้าพวกนั้นเป็น 30 วัน**

**(0?[1-9]|1[0-9]|2[0-8])[/]0?2)[/]?[0-9]{1,4}29[/]0?2[/]?([0-9]{2} หมายถึง ในเดือน 2 นั้นสามารถมี 29 วันได้ซึ่งจะถูกตรวจสอบกับปีว่าตรงกับ leap year หรือไม่หากไม่ก็จะมีเพียง 28 วัน ซึ่งการกรอกปี นั้นสามารถใส่ได้ 1-9999 ปีใน code นี้สามาถทำได้**

**(([2468][048]|[02468][48])|[13579][26])|([13579][26]|[02468][048]|0[0-9]|1[0-6])00))$ เป็นการทำการตรวจสอบ leap year day**

****

***(****ผลลัพธ์จากการกรองวัน)*