

2. [ 6 คะแนน ] จงออกแบบแล็กอริทึมในการหาลำดับย่อย (subsequence) ของลำดับของตัวเลขขนาด  $n$  จำนวน ที่ลำดับย่อยนั้นมีผลบวกเท่ากับ  $K$  ที่กำหนดมาให้ ทั้งนี้ ในลำดับของตัวเลขอาจมีทั้งจำนวนเต็มบวก ศูนย์ และจำนวนเต็มลบได้
3. [ 6 คะแนน ] จงออกแบบอัลกอริทึมสำหรับปัญหาการหา Maximum Contiguous Subsequence Sum คือ กำหนดให้  $a_1, a_2, \dots, a_n$  เป็นลำดับของจำนวนเต็ม (บวก ลบ หรือ ศูนย์)  $n$  จำนวน ต้องการหาว่า ค่ามากที่สุดที่เป็นไปได้ของ  $\text{sum} = \sum_{k=i \text{ to } j} a_k$  สำหรับทุกค่าของ  $i$  ถึง  $j$  คือเท่าใด (ในกรณีที่ทุกจำนวนเป็นจำนวนเต็มลบ ให้ค่า  $\text{sum}$  เป็นศูนย์ ซึ่งหมายถึง empty subsequence)
4. [ 6 คะแนน ] กำหนดให้มีสายอักขระสองสาย  $A[n]$  และ  $B[m]$  ขนาด  $n$  และ  $m$  ตามลำดับ จงออกแบบอัลกอริทึมในการหา longest common substring ของ  $A$  และ  $B$  ที่กำหนดมาให้

5. [ 6 คะแนน ] กำหนดให้  $B$  คือ อาร์เรย์ขนาด  $n \times n$  ซึ่งเก็บเฉพาะเลข 1 กับ 0 โดยที่ในแต่ละแถวแนวนอนของ  $B$  เลข 1 ใดๆ ต้องมาก่อนเลข 0 (จากซ้ายมาขวา)

	0	1	2	3	4	5
0	1	1	1	0	0	0
1	1	0	0	0	0	0
2	1	1	1	1	0	0
3	1	1	0	0	0	0
4	1	0	0	0	0	0
5	1	1	0	0	0	0

- 5.1 จงออกแบบอัลกอริทึมที่หาว่าแถวแนวนอนแถวใดใน  $B$  ที่มีจำนวนของเลข 1 มากที่สุด (หาเพียงแถวเดียวก็พอ ในกรณีที่有多จำนวนมากที่สุดเท่ากันหลายแถว) ตัวอย่างเช่นแถวแนวนอนหมายเลข 2 ของอาร์เรย์  $B$  ข้างล่างนี้มีจำนวนเลข 1 มากที่สุด โดยมีความซับซ้อนเชิงเวลาเป็น  $O(n)$  เมื่อ  $n$  คือจำนวนแถว (หรือคอลัมน์)
- 5.2 ทดสอบอัลกอริทึมกับตัวอย่างที่ให้ และนับจำนวนการเปรียบเทียบ ว่ามีทั้งหมดเท่าไร (เทียบค่าในอาร์เรย์ว่าเป็น 0 หรือ 1)