ภาคเรียนที่ 2 ปีการศึกษา 2562 มหาวิทยาลัยธรรมศาสตร์ โดย คณาจารย์วิชา คพ.300

#### หีบแห่งคพ300 (TheArkOfCS300)

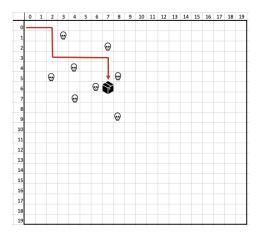
หีบแห่งคพ300 (TheArkOfCS300) เรียกได้ว่าเป็นหีบศักดิ์สิทธิ์สำหรับนักศึกษาทางคอมพิวเตอร์ทีเดียว ทางสายมูเตลูเชื่อ กันว่ากูรูทางด้านคอมพิวเตอร์ได้รวมตัวกันสร้างหีบนี้ขึ้นเพื่อผนึกศาสตร์ความรู้ทางด้านคอมพิวเตอร์ไว้ภายใน ใครที่เข้าถึงและได้ ครอบครองหีบใบนี้ ก็จะเป็นผู้เชี่ยวชาญทางคอมพิวเตอร์ที่หาใครเทียบไม่ได้ ดังนั้นหีบศักดิ์สิทธิ์ใบนี้จึงเป็นที่หมายปองของนักศึกษา คอมพิวเตอร์ทั้งหลาย หีบแห่งคพ300นี้มีไม่จำกัด นักศึกษาที่มีการเตรียมตัวมาอย่างดีจะสามารถได้หีบใบนี้ไปครอบครองทุกคน ครั้งนี้ นักศึกษาได้ไล่ล่าและตามหาหีบใบนี้จนมาพบว่ามันถูกซ่อนไว้ที่ห้องแห่งหนึ่งที่ตึกบร.2 แต่การจะเข้าไปถึงหีบ คพ300 ได้นั้น จะต้อง วางแผนการเดินโดยเขียนเป็นแผนผังมาก่อนเป็นรหัส แล้วเดินไปตามเส้นทางทีละย่างกาวอย่างระมัดระวัง เพราะภายในห้องนั้นมีกับดัก อยู่เพียบ การกาวเดินภายในห้องนั้นสามารถเดินได้ 4 ทิศรอบตัวเท่านั้น และจะเป็นการเดินทีละกาว ถ้ากาวไปยังตำแหน่งที่มีกับดักอยู่ ก็จะถือว่าสิ้นสุดการเดินไม่สามารถเข้าไปถึงหีบสมบัติได้

จงเขียนโปรแกรมจำลองการเดินเพื่อเข้าไปเอาหีบแห่งคพ300 โดยกำหนดให้ห้องมีขนาด  $20 \times 20$  ช่อง กำหนดทิศทางการ ก้าวไปทั้งแปดทิศด้วยตัวเลข 1-4 ดังแสดงในรูปที่ 1

	1	
4	序	2
	3	

รูปที่ 1 รหัสทิศทั้ง 4 ทิศที่ใช้กำหนดในการเดิน

ถ้ากำหนดตำแหน่งของห้องขนาด 20 × 20 และการระบุตำแหน่งเป็นคู่ลำดับของแถว i กับคอลัมน์ j ตามรูปที่ 2 โปรแกรมจะ รับค่าตำแหน่งที่วางหีบ (i, j) พร้อมทั้งระบุว่ามีกับดักอยู่จำนวนเท่าใดและที่ตำแหน่งใหนในห้องโดยระบุเป็นคู่ลำดับของตำแหน่งในห้อง จากนั้นจะระบุว่านักศึกษาจะเดินกี่ก้าว และระบุเส้นทางการเดินของนักศึกษาโดยใช้รหัสทิศทั้งสี่ใส่ตามลำดับ โดยนักศึกษาจะเริ่มต้น โดยอยู่ที่ตำแหน่ง (0, 0) ก่อนการเริ่มต้นเดินเสมอ ในการเดินแต่ละครั้งโปรแกรมก็จะต้องคำนวณหาตำแหน่งใหม่ที่นักศึกษาจะก้าวไป อยู่ ถ้าตำแหน่งใหม่เป็นตำแหน่งที่อยู่ในขอบเขตของห้องนักศึกษาก็จะเคลื่อนย้ายไปยังตำแหน่งใหม่ แต่ถ้าตำแหน่งใหม่เป็นตำแหน่งที่ ออกนอกขอบเขตห้อง นักศึกษาก็จะต้องอยู่กับที่ไม่สามารถเคลื่อนย้ายไปไหนได้ และถ้าเดินไปอยู่ในตำแหน่งที่มีกับดักอยู่การเดินก็ จะสิ้นสุดทันที และถือว่าไม่สามารถเดินไปถึงหีบแห่งคพ300ได้ แต่ถ้าไม่เจอกับดักก็จะต้องเดินไปเรื่อย ๆ จนสิ้นสุดเส้นทางที่วางไว้ เมื่อ สิ้นสุดการเดินโปรแกรมจะต้องรายงานตำแหน่งสุดท้ายที่นักศึกษาอยู่ และระบุว่าภารกิจสำเร็จหรือไม่ ถ้าตำแหน่งสุดท้ายเป็นตำแหน่งที่ อยู่ติดกับหีบ (ในทิศทาง 1-4) หรือตำแหน่งของหีบ จะถือว่าภารกิจสำเร็จ แต่ถ้าไม่ใช่ตำแหน่งดังกล่าวก็จะถือว่าหาไม่เจอภารกิจล้มเหลว



รูปที่ 2 ตัวอย่างตำแหน่งของหีบแห่งคพ300 กับดับ และเส้นทางการเดิน

ตัวอย่างเช่น หีบวางอยู่ที่ตำแหน่ง (6, 7) และมีกับดัก 8 จุดคือ (1, 3), (9, 8), (5, 2), (4, 4), (5, 8), (2, 7), (6, 6), และ (7, 4) นักศึกษาวางแผนการเดินดังนี้ [2 2 3 3 3 2 2 2 2 2 3 3] ดังแสดงในรูปที่ 2

ดังนั้นจุดสุดท้ายที่นักศึกษาอยู่เมื่อสิ้นสุดการเดินคือ (5, 7) และภารกิจสำเร็จ

### ข้อมูลเข้า

กำหนดให้ขนาดของห้องเท่ากับ  $20 \times 20$  หีบสมบัติวางที่ตำแหน่ง (i,j) และมีกับดักอยู่ K จุด นักศึกษาวางแผนการเดิน M ก้าว ดังนั้นข้อมูลที่นำเข้าจะมี K+3 บรรทัด

- บรรทัดที่ 1 มีจำนวนเต็มบวกสามจำนวน <u>แต่ละจำนวนคั่นด้วยช่องว่างหนึ่งช่อง</u>
  - O จำนวนที่หนึ่งและสองระบุตำแหน่ง i และ j แทนตำแหน่งที่หีบสมบัติวางอยู่
  - $\circ$  จำนวนที่สามระบุถึงจำนวนกับดัก K จุดที่มีในห้อง
- บรรทัดที่ 2 ถึง K + 1 มีจำนวนเต็มบวกสองจำนวนระบุตำแหน่งของกับดัก โดยแต่ละจำนวนคั่นด้วยช่องว่างหนึ่งช่อง
  จำนวนแรกระบุแถว และจำนวนที่ 2 ระบุคอลัมน์
- บรรทัดที่ K+2 จะบอกจำนวนก้าวที่นักศึกษาจะเดิน M ก้าว ( $1 \le M \le 1000$ )
- สำหรับบรรทัดที่ K + 3 จะมีตัวเลข M เป็นตัวเลขจำนวนเต็มที่มีค่า 1 4 แทนทิศทางการก้าวเดินขอนักศึกษาแต่ละก้าว ตามที่โจทย์กำหนด

#### <u>หมายเหตุ</u>

กำหนดให้ข้อมูลเข้าทุกตัวมีค<sup>่</sup>าถูกต้องตามรูปแบบ ขอบเขต และเซ็ตของค<sup>่</sup>าที่เป็นไปได้เสมอ นักศึกษาไม่จำเป็นต้องตรวจสอบ (validate) ข้อมูลเข้า

### ข้อมูลส่งออก

มี 2 บรรทัด โดย

- บรรทัดแรก แสดงจุดสุดท้ายของการเดิน โดยให้แสดงเป็นตำแหน่งแถว ( *i* ) และคอลัมน์ ( *j* ) บนกระดานตามลำดับ ซึ่ง แทนด้วยเลขจำนวนเต็มบวก 2 จำนวน คั่นแต่ละจำนวนด้วยช<sup>่</sup>องว่างหนึ่งช<sup>่</sup>อง
- บรรทัดที่สอง แสดงผลลัพธ์เป็น 1 ถ้าภารกิจสำเร็จ และเป็น 0 ถ้าภารกิจล้มเหลว

**ตัวอย่างที่ 1** ถ้าหีบวางอยู่ที่ตำแหน่ง (6, 7) และมีกับดัก 8 จุดคือ (1, 3), (9, 8), (5, 2), (4, 4), (5, 8), (2, 7), (6, 6), และ (7, 4) นักศึกษาวางแผนการเดินดังนี้ [2 2 3 3 3 2 2 2 2 2 3 3]

ข้อมูลเข้า	ข้อมูลส่งออก
6 7	5 7
8	1
1 3	
98	
5 2	
4 4	
5 8	
2 7	
6 6	
7 4	
12	
22333222233	

**ตัวอย่างที่ 2** ถ้าหีบวางอยู่ที่ตำแหน่ง (6, 7) และมีกับดัก 8 จุดคือ (1, 3), (9, 8), (5, 2), (4, 4), (5, 8), (2, 7), (6, 6), และ (7, 4) นักศึกษาวางแผนการเดินดังนี้ [2 2 2 2 1 2 2 2 3 3 3 3 3]

ข้อมูลเข้า	ข้อมูลส่งออก
6 7	2 7
8	0
1 3	
9 8	
5 2	
4 4	
5 8	
2 7	
6 6	
7 4	
13	
2 2 2 2 1 2 2 2 3 3 3 3 3	

**ตัวอย่างที่ 3** ถ้าหีบวางอยู่ที่ตำแหน่ง (6, 7) และมีกับดัก 8 จุดคือ (1, 3), (9, 8), (5, 2), (4, 4), (5, 8), (2, 7), (6, 6), และ (7, 4) นักศึกษาวางแผนการเดินดังนี้ [2 2 3 3 3 2 2 2 2 2 2 1 1 4 4 4 4 5 5 5 5 5]

ข้อมูลเข้า	ข้อมูลส่งออก
6 7	6 5
8	0
1 3	
9 8	
5 2	
4 4	
5 8	
2 7	
6 6	
7 4	
23	
223333222222114444333333	

## ข<sup>้</sup>อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อ
	จากผลลัพธ์สุดท้าย
เงื่อนไขในการให้คะแนน	โปรแกรมจะต้องประมวลผลชุดข้อมูลทดสอบที่ผู้ตรวจเตรียมไว้ได้ถูกต้อง

# ข<sup>้</sup>อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมไพเลอร์ที่ส่วนหัวของโปรแกรมดังนี้

ภาษา C และ MinGW 4.4.1	ภาษา C++ และ MinGW 4.4.1
(Code::Blocks บนวินโดวส์)	(Code::Blocks บนวินโดวส์)
/*	/*
LANG: C	LANG: C++
COMPILER: WCB	COMPILER: WCB
*/	*/
ภาษา C และ MinGW 3.4.2	ภาษา C++ และ MinGW 3.4.2
(Dev-C++ บนวินโดวส์)	(Dev-C++ บนวินโดวส์)
/*	/*
LANG: C	LANG: C++
COMPILER: WDC	COMPILER: WDC
*/	*/
ภาษาจาวา และ jdk1.7.0_71	
/*	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็น
LANG: JAVA	ชื่อเดียวกับโจทย์ และไม่มี
COMPILER: JAVA	การสร้างแพคเกจย่อย
*/	ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp
	หรือ .java