



แบบฝึกหัดภาคปฏิบัติการ
วิชา 51711 การเขียนโปรแกรมคอมพิวเตอร์ 1

ภิญโญ แท้ประสาทสิทธิ์

ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร 2559

สารบัญเนื้อหา

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่หนึ่ง ชุดที่หนึ่ง พื้นฐานการวิเคราะห์และแก้ปัญหา.....	6
ปัญหา ปลายทาง.....	6
ปัญหา สมัครสมาชิก.....	8
ปัญหา ขับรถตามให้ทัน.....	10
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่หนึ่ง ชุดที่สอง พื้นฐานการวิเคราะห์และแก้ปัญหา.....	12
ปัญหา จองรถตู้ [reserve_van_shift].....	13
ปัญหา วันที่ตัดเสื้อแล้วเสร็จ [due_date].....	16
ปัญหา คัดเลือกนักร้อง [the_voice_su].....	19
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สอง การวางแผนการคำนวณด้วยโฟลวชาร์ตและซูโดโค้ด: การใช้โครงสร้างตัดสินใจ (IF).....	22
แบบฝึกหัดชุดที่ 1: การรับข้อมูลเข้า การคำนวณ และการแสดงผล.....	22
แบบฝึกหัดชุดที่ 2: การแยกกรณีการทำงานด้วยเงื่อนไข (condition) แบบ IF แบบพื้นฐาน.....	22
แบบฝึกหัดชุดที่ 3: การแยกกรณีการทำงานด้วยเงื่อนไข (condition) แบบ IF-ELSE.....	22
แบบฝึกหัดชุดที่ 4: การแยกกรณีการทำงานด้วยเงื่อนไข (condition) แบบ IF และ ELSE IF.....	23
แบบฝึกหัดชุดที่ 5: ความเทียบเท่ากันของการใช้เงื่อนไขแบบต่าง ๆ.....	24
แบบฝึกหัดชุดที่ 6: หัดเลือกใช้ IF ที่เหมาะสม.....	25
แบบฝึกหัดชุดที่ 7: การใช้เงื่อนไขที่ซับซ้อนในการแก้ปัญหา.....	27
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สาม การวางแผนการคำนวณด้วยโฟลวชาร์ตและซูโดโค้ด: การใช้โครงสร้างการทำซ้ำ.....	29
แบบฝึกหัดชุดที่ 1: ฝึกการวนซ้ำเพื่อนับจำนวนที่ต้องการ.....	29
แบบฝึกหัดชุดที่ 2: ฝึกการวนซ้ำกับปัญหาที่ซับซ้อนขึ้น.....	30
แบบฝึกหัดชุดที่ 3: การวนซ้ำในปัญหาที่จำลองจากเหตุการณ์จริง.....	32
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สี่ เตรียมตัวสอบย่อยด้วยตัวอย่างข้อสอบย่อยปีการศึกษา 2556 ภาคการศึกษา ต้น.....	33
เฉลยข้อสอบย่อยครั้งที่หนึ่ง วิชา 51711 Computer Programming I.....	37
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่ห้า พื้นฐานการเขียนภาษาซี เงื่อนไขการทำงาน.....	39
ปัญหาชุดที่ 1: พื้นฐานการรับข้อมูลเข้า การคำนวณเลขคณิต และการแสดงผล.....	39
ปัญหา 1 เกี่ยวกับตัวคุณ [about_you].....	39
ปัญหา 2 พิมพ์ฉัน [print_me].....	39
ปัญหา 3 พิมพ์ฉันภาคสอง [print_me2].....	40
ปัญหา 4 บวกเลขจำนวนเต็ม [add_number].....	40
ปัญหา 5 เลขหลักหน่วย [last_digit].....	41
ปัญหา 6 เลขหลักสิบ [second_last_digit].....	41

ปัญหาชุดที่ 2: รู้จักกับจำนวนและการเปรียบเทียบค่าในคำสั่งเงื่อนไข.....	41
ปัญหา 7 เลขบวก [positive].....	42
ปัญหา 8 เลขคู่หรือคี่ [even_or_odd].....	42
ปัญหา 9 บวก ลบ หรือว่า ศูนย์ [number_sign].....	42
ปัญหา 10 อย่าหารเลขด้วยศูนย์ [no_divide_by_zero].....	42
ปัญหา 11 หาเศษและพิมพ์ผลลัพธ์ [modulo_and_printf].....	43
ปัญหา 12 เลือกตัวดำเนินการ [operator_selection].....	43
ปัญหา 13 ค่าสัมบูรณ์ [absolute_value].....	43
ปัญหาชุดที่ 3: การประยุกต์ใช้การดำเนินการคณิตศาสตร์ในการแก้ปัญหา.....	44
ปัญหา 14 ตัดเกรดสามระดับ [grading_fpe].....	44
ปัญหา 15 เรียงเบอร์สามตัว [3num_sort].....	44
ปัญหา 16 จัดถุงยังชีพ [survival_bag].....	45
ปัญหา 17 จองรถตู้ [reserve_van].....	45
ปัญหา 18 คัดเลือกนักร้อง [the_voice_su].....	46
ปัญหา 19 รับคำสั่งซื้อลูกค้าได้หรือไม่ [consider_order].....	47
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่หก การวนซ้ำและการจัดรูปแบบการแสดงผล.....	49
ปัญหาชุดที่ 1 หัดใช้โครงสร้างการวนซ้ำ.....	49
ปัญหา 1 วนไปพิมพ์ไป [print_loop].....	49
ปัญหา 2 วนไปพิมพ์ไปภาคสอง [print_loop2].....	50
ปัญหา 3 วนไปในขอบเขตที่กำหนด [loop_range].....	50
ปัญหา 4 วนไปในขอบเขตที่กำหนดภาคสอง [loop_range2].....	50
ปัญหาชุดที่ 2 เทคนิคพื้นฐาน.....	51
ปัญหา 5 จำนวนบวกที่น้อยและมากที่สุด [positive_min_max].....	51
ปัญหา 6 คู่หรือคี่ใครแน่จริง [odd_vs_even].....	53
ปัญหา 7 หาจำนวนที่มีค่าใกล้เคียงที่สุดกับเลขที่กำหนด [closest_number].....	54
ปัญหาชุดที่ 3 การนับจำนวน.....	56
ปัญหา 8 หยุดลูปโดยพลัน [how_to_break].....	56
ปัญหา 9 นับจำนวนที่สนใจ [count_target_number].....	57
ปัญหา 10 นับจำนวนเลขซ้ำติดกันที่เราสนใจ [target_max_consec].....	58
ปัญหา 11 นับจำนวนเลขซ้ำติดกันที่ยาวที่สุด [any_max_consec].....	59
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่เจ็ด การวนซ้ำและการจัดรูปแบบการแสดงผล 2.....	60
ปัญหา 1 วางแผนผลิตชาเขียว [green_tea].....	60
ปัญหา 2 วันที่ตัดเสื้อแล้วเสร็จ [due_date].....	61
ปัญหา 3 จองรถตู้ [reserve_3_vans].....	63

ปัญหา 4 คัดเลือกนักร้อง [the_voice_su].....	65
ปัญหา 1 จัดตารางสูตรคูณ.....	67
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่แปด การซ้อนลูป.....	68
ปัญหา 1 กระจหน้าพิมพ์ดอกจัน [AsteriskBurst].....	68
ปัญหา 2 นับเลขแบบต่อแถว [RowCounting].....	68
ปัญหา 3 พิมพ์พิกัดของแถวและคอลัมน์ [RCCoord].....	69
ปัญหา 4 กราฟแท่งแนวนอน [HorizontalBarGraph].....	69
ปัญหา 5 ระบายพื้นที่สี่เหลี่ยมขนมเปียกปูน [Rhombus].....	70
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่เก้า อาเรย์หนึ่งมิติ.....	71
ปัญหา 1 พิมพ์เลขย้อนกลับ [ReversePrint].....	71
ปัญหา 2 การแข่งขันประเภททีม [TeamCompetition].....	71
ปัญหา 3 ตำแหน่งของเลขที่สนใจ [NumberOccurrence].....	73
ปัญหา 4 บันไดตัวเลข (NumberStairs2).....	74
ปัญหา 5 พิมพ์กราฟแท่งแนวตั้ง เวอร์ชันพื้นฐาน (vertical_bar_graph_basic).....	75
ปัญหา 6 หาดที่ตั้งร้านขายของริมทาง (road_side_shop).....	76
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สิบ อาเรย์สองมิติ.....	77
ปัญหา 1 การทรานส์โพสเมตริกซ์ [transpose_matrix].....	77
ปัญหา 2 ผลรวมความแตกต่างของเมตริกซ์ [sum_matrix_diff].....	78
ปัญหา 3 นับยอดเขา [mountain_top].....	79
ปัญหา 4 พิมพ์แผนที่ลานจอดรถ [car_park_map].....	81
ปัญหา 5 จุดอานม้า [saddle_point].....	83
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สิบเอ็ด ฟังก์ชันและตัวชี้.....	84
ปัญหา 1 ค่าเฉลี่ยและความแปรปรวน [average_and_variance1].....	84
ปัญหา 2 ค่าเฉลี่ยและความแปรปรวน [average_and_variance2].....	85
ปัญหา 3 ฟังก์ชันทดสอบว่ามีเลขที่ซ้ำกันอยู่ในอาเรย์หรือไม่ [isUnique].....	86
ปัญหา 4 อนุพันธ์ในอาเรย์ [array_derivative].....	87
ปัญหา 5 ฟังก์ชันนับการซ้ำของเลขที่มีค่าสูงสุดในอาเรย์ [count_max].....	88
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สิบสอง เตรียมสอบอาเรย์.....	90
ปัญหา 1 พิมพ์เลขเดินหน้าหรือย้อนหลัง [forward_backward_print].....	90
ปัญหา 2 นับคะแนนเลือกตั้ง [count_vote].....	91
ปัญหา 3 แผนที่บอกลำดับเส้นทางอย่างง่าย [waypoint_order].....	92
แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สิบสาม สตริง.....	94
ปัญหา 1 บันทึกและค้นข้อมูลนักศึกษา [data_record_and_retrieval].....	94
ปัญหา 2 คะแนนเฉลี่ยของผู้เข้าสอบ [average_score].....	96

ปัญหา 3 จำนวน รหัส และ รายชื่อของผู้ที่เข้ารอบ [qualified_examinees].....	97
ปัญหา 4 หาผู้ได้คะแนนรวมสูงสุด [find_top_student].....	99

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่หนึ่ง ชุดที่หนึ่ง

พื้นฐานการวิเคราะห์และแก้ปัญหา

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

บทนำ

การวิเคราะห์และแก้ปัญหาเป็นพื้นฐานที่สำคัญมากในการเขียนโปรแกรม เนื่องจากผู้เขียนโปรแกรมจะต้องสามารถวิเคราะห์และแก้ปัญหาได้ด้วยตนเองเสียก่อนจึงจะสามารถเขียนโปรแกรมได้ ยิ่งไปกว่านั้น เราจำเป็นที่จะต้องเข้าใจวิธีการแก้ปัญหาโดยชัดเจนและอธิบายมันออกมาได้ เพราะการเขียนโปรแกรมนั้นแท้จริงก็คือการบอกคอมพิวเตอร์ให้คิดตามเรานั้นเอง หากเราวิเคราะห์และแก้ปัญหาไม่ได้ หรือเราทำได้แต่อธิบายไม่ได้ เราก็จะไม่สามารถเขียนโปรแกรมให้สำเร็จได้

แบบฝึกหัดนี้มีจุดประสงค์เพื่อให้ผู้เรียนได้รับรู้ถึงระดับความสามารถในการวิเคราะห์และแก้ปัญหาของตนเอง พร้อมทั้งแสดงให้เห็นว่า 'แม้ในปัญหาเดียวกันก็สามารถพลิกออกมาได้หลายรูปแบบ แต่หากเข้าใจอย่างชัดเจนแล้วไม่ว่าปัญหาจะถูกพลิกออกมาในรูปแบบใด ก็ไม่ได้เป็นอุปสรรคอะไรกับเราเลย'

ปัญหา ปลากระบ้อง

(แนะนำให้ทำข้อ 1, 3, 7, 9, 13, 17 และ 20)

Level 1 : แบบพื้นฐาน (Basic)

1. ปลากระบ้องยี่ห้อหนึ่งใช้ปลาซาร์ดีนสามตัวและมะเขือเทศสองผลเพื่อผลิตปลากระบ้องหนึ่งกระบ้อง หากโรงงานผลิตมีปลาซาร์ดีนอยู่ 300 ตัวและมะเขือเทศอยู่ 200 ผล โรงงานจะผลิตปลากระบ้องได้ทั้งหมดกี่กระบ้อง
2. จากข้อหนึ่ง ถ้าหากโรงงานมีปลาซาร์ดีนอยู่ 600 ตัวและมะเขือเทศ 200 ผล โรงงานจะผลิตปลากระบ้องได้ทั้งหมดกี่กระบ้อง
3. จากข้อหนึ่ง ถ้าหากโรงงานมีปลาซาร์ดีนอยู่ 520 ตัวและมะเขือเทศ 333 ผล โรงงานจะผลิตปลากระบ้องได้ทั้งหมดกี่กระบ้อง
4. จากข้อหนึ่ง ถ้าหากโรงงานมีปลาซาร์ดีนอยู่ 700 ตัวและมะเขือเทศ 500 ผล โรงงานจะผลิตปลากระบ้องได้ทั้งหมดกี่กระบ้อง
5. จากข้อหนึ่ง ถ้าหากโรงงานมีปลาซาร์ดีนอยู่ 900 ตัวและมะเขือเทศ 100 ผล โรงงานจะผลิตปลากระบ้องได้ทั้งหมดกี่กระบ้อง
6. จากข้อหนึ่ง ถ้าหากโรงงานมีปลาซาร์ดีนอยู่ 100 ตัวและมะเขือเทศ 500 ผล โรงงานจะผลิตปลากระบ้องได้ทั้งหมดกี่กระบ้อง

Level 2 : แบบหลายจุดประสงค์ (Multi-objective)

คำถามชุดนี้ต้องการทำให้ผู้เรียนคุ้นเคยกับปัญหาที่ต้องการคำตอบมากกว่าหนึ่งอย่าง และสังเกตเห็นรายละเอียดปลีกย่อยของปัญหา และเข้าใจความสัมพันธ์ของข้อมูลต่าง ๆ อย่างชัดเจน

7. ปลากระป๋องยี่ห้อหนึ่งใช้ปลาซาร์ดีนสามตัวและมะเขือเทศสองผลเพื่อผลิตปลากระป๋องหนึ่งกระป๋อง หากโรงงานผลิตมีปลาซาร์ดีนอยู่ 300 ตัวและมะเขือเทศอยู่ 200 ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง และจะมีวัตถุดิบแต่ละอย่างคงเหลือเท่าใดบ้าง
8. จากข้อ 7 ถ้าหากโรงงานมีปลาซาร์ดีนอยู่ 600 ตัวและมะเขือเทศ 200 ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง และจะมีวัตถุดิบแต่ละอย่างคงเหลือเท่าใดบ้าง
9. จากข้อ 7 ถ้าหากโรงงานมีปลาซาร์ดีนอยู่ 520 ตัวและมะเขือเทศ 333 ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง และจะมีวัตถุดิบแต่ละอย่างคงเหลือเท่าใดบ้าง
10. จากข้อ 7 ถ้าหากโรงงานมีปลาซาร์ดีนอยู่ 700 ตัวและมะเขือเทศ 500 ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง และจะมีวัตถุดิบแต่ละอย่างคงเหลือเท่าใดบ้าง
11. จากข้อ 7 ถ้าหากโรงงานมีปลาซาร์ดีนอยู่ 900 ตัวและมะเขือเทศ 100 ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง และจะมีวัตถุดิบแต่ละอย่างคงเหลือเท่าใดบ้าง
12. จากข้อ 7 ถ้าหากโรงงานมีปลาซาร์ดีนอยู่ 100 ตัวและมะเขือเทศ 500 ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง และจะมีวัตถุดิบแต่ละอย่างคงเหลือเท่าใดบ้าง

Level 3 : แบบเชิงสัญลักษณ์ (Symbolic)

คำถามชุดนี้เป็นตัวจริงที่ผู้เรียนต้องผ่านไปให้ได้ เพราะในโปรแกรมมักจะมีค่าที่ต้องนำมาจากผู้ใช้ ค่าจากผู้ใช้นี้มีความประหลาดเป็นเหมือนตัวแปรไม่ทราบค่าในสมการ และมันมักจะเป็นตัวแปรในโปรแกรมของเราด้วย ดังนั้นผู้เรียนจึงต้องเข้าใจการคำนวณโดยติดตัวแปรไม่ทราบค่านี้ไว้ และสามารถเขียนอธิบายคำตอบสุดท้ายจากตัวแปรนี้ได้ (คือจะต้องตอบคำถามออกมาในรูปติดตัวแปร X , Y , M , และ/หรือ N เช่น ได้จำนวนกระป๋องเป็น $X + 10$)

คำตอบในคำถามชุดนี้จะติดตัวแปร X , Y , M , หรือ N ไว้ และผู้เรียนจะต้องแบ่งกรณีคำตอบออกเป็นหลาย ๆ แบบ เพื่อให้คำตอบครอบคลุมทุกความเป็นไปได้ ผู้เรียนควรทำข้อ 13 และ 14 ให้ได้เป็นอย่างน้อย

ในกรณีที่รู้สึกสงสัยว่าคำตอบที่แบ่งออกเป็นหลายกรณีนี้ครอบคลุมทุกความเป็นไปได้แล้วหรือยัง ให้เราลองคิดทบทวนไปที่สิ่งที่เราทำในข้อ 1-6

13. ปลากระป๋องยี่ห้อหนึ่งใช้ปลาซาร์ดีนสามตัวและมะเขือเทศสองผลเพื่อผลิตปลากระป๋องหนึ่งกระป๋อง หากโรงงานผลิตมีปลาซาร์ดีนอยู่ X ตัวและมะเขือเทศอยู่ 200 ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง
14. จากข้อ 13 ถ้าหากโรงงานผลิตมีปลาซาร์ดีนอยู่ 300 ตัวและมะเขือเทศอยู่ Y ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง

15. จากข้อ 13 ถ้าหากโรงงานผลิตมีปลาซาร์ดีนอยู่ X ตัวและมะเขือเทศอยู่ Y ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง

16. จากข้อ 13 ถ้าหากโรงงานผลิตมีปลาซาร์ดีนอยู่ X ตัวและมะเขือเทศอยู่ Y ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง และจะมีวัตถุดิบแต่ละอย่างคงเหลือเท่าใดบ้าง

17. ปลากระป๋องยี่ห้อหนึ่งใช้ปลาซาร์ดีน M ตัวและมะเขือเทศสองผลเพื่อผลิตปลากระป๋องหนึ่งกระป๋อง หากโรงงานผลิตมีปลาซาร์ดีนอยู่ 600 ตัวและมะเขือเทศอยู่ 200 ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง

18. ปลากระป๋องยี่ห้อหนึ่งใช้ปลาซาร์ดีนสามตัวและมะเขือเทศ N ผลเพื่อผลิตปลากระป๋องหนึ่งกระป๋อง หากโรงงานผลิตมีปลาซาร์ดีนอยู่ 600 ตัวและมะเขือเทศอยู่ 200 ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง

19. ปลากระป๋องยี่ห้อหนึ่งใช้ปลาซาร์ดีน M ตัวและมะเขือเทศ N ผลเพื่อผลิตปลากระป๋องหนึ่งกระป๋อง หากโรงงานผลิตมีปลาซาร์ดีนอยู่ 600 ตัวและมะเขือเทศอยู่ 200 ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง

Level 4 : แบบขั้นสูงสุด (Ultimate)

หากผู้เรียนเข้าใจวิธีคิดทุกอย่างโดยชัดเจนแล้ว ผู้เรียนจะสามารถคิดคำตอบข้อนี้ออกมาได้ ซึ่งจะมีกรณีและความเป็นไปได้ที่ซับซ้อน ชวนงุนงง และง่ายที่จะเกิดความผิดพลาดในการหาคำตอบ หรือคิดกรณีที่ซ้ำซ้อนฟุ่มเฟือยออกมา

20. ปลากระป๋องยี่ห้อหนึ่งใช้ปลาซาร์ดีน M ตัวและมะเขือเทศ N ผลเพื่อผลิตปลากระป๋องหนึ่งกระป๋อง หากโรงงานผลิตมีปลาซาร์ดีนอยู่ X ตัวและมะเขือเทศอยู่ Y ผล โรงงานจะผลิตปลากระป๋องได้ทั้งหมดกี่กระป๋อง และจะมีวัตถุดิบแต่ละอย่างคงเหลือเท่าใดบ้าง

ปัญหา สมัครสมาชิก

(แนะนำให้ทำข้อ 1, 5, 7 และ 9)

Level 1 : แบบพื้นฐาน (Basic)

1. ร้านอินเทอร์เน็ตคาเฟ่ร้านหนึ่งคิดค่าใช้บริการ 15 บาทต่อชั่วโมง จงคำนวณค่าใช้จ่ายของลูกค้าที่ใช้บริการเป็นเวลา 50 ชั่วโมง

2. ร้านอินเทอร์เน็ตคาเฟ่ร้านหนึ่งคิดค่าใช้บริการ 15 บาทต่อชั่วโมง จงคำนวณค่าใช้จ่ายของลูกค้าที่ใช้บริการเป็นเวลา 100 ชั่วโมง

Level 2 : แบบหลายจุดประสงค์ (Multi-Objective)

3. ร้านอินเทอร์เน็ตคาเฟ่ร้านหนึ่งคิดค่าใช้บริการ 15 บาทต่อชั่วโมง ทางร้านต้องการดึงดูดลูกค้าที่ใช้อินเทอร์เน็ตหรือเล่นเกมส์มาก จึงมีระบบสมาชิกร้านซึ่งผู้สมัครสมาชิกจะเสียค่าสมาชิก 100 บาทต่อเดือนและจะได้ส่วนลด 10% จากค่าบริการ จงคำนวณค่าใช้จ่ายทั้งหมดต่อเดือนระหว่างผู้ใช้บริการ A และ B โดยที่ทั้ง A และ B ต่างก็ใช้บริการ 50

ชั่วโมงต่อเดือนที่ร้านทั้งคู่ โดยที่ A ไม่เป็นสมาชิก และ B เป็นสมาชิก (ค่าใช้จ่ายทั้งหมดของ B ต้องรวมค่าสมาชิกด้วย) และสรุปว่าใครเสียค่าใช้จ่ายมากกว่ากัน

4. จากข้อหนึ่ง จงคำนวณค่าใช้จ่ายทั้งหมดต่อเดือนระหว่างผู้ใช้บริการ C และ D โดยที่ทั้ง C และ D ต่างก็ใช้บริการ 100 ชั่วโมงต่อเดือนที่ร้านทั้งคู่ โดยที่ C ไม่เป็นสมาชิก และ D เป็นสมาชิก (ค่าใช้จ่ายทั้งหมดของ D ต้องรวมค่าสมาชิกด้วย) และสรุปว่าใครเสียค่าใช้จ่ายมากกว่ากัน

5. ร้านอินเทอร์เน็ตคาเฟ่ร้านหนึ่งคิดค่าใช้จ่ายค่าบริการ 15 บาทต่อชั่วโมง ทางร้านต้องการดึงดูดลูกค้าที่ใช้อินเทอร์เน็ตหรือเล่นเกมส์มาก จึงมีระบบสมาชิกร้านซึ่งผู้สมัครสมาชิกจะเสียค่าสมาชิก 100 บาทต่อเดือนและจะได้ส่วนลด 10% จากค่าบริการ โดยเฉลี่ยแล้วเด็กชายต้นจะไปใช้บริการที่ร้านอินเทอร์เน็ตทุกวัน วันละสองชั่วโมง สมมติให้แต่ละเดือนมี 30 วัน จงหาว่าเด็กชายต้นควรจะสมัครสมาชิกร้านอินเทอร์เน็ตนี้หรือไม่

(ข้อนี้ถึงแม้คำตอบสุดท้ายต้องการคำตอบเดียว แต่ผู้เรียนจะเห็นได้ว่าระหว่างทางสู่คำตอบสุดท้ายนั้น เราจะต้องทำการคำนวณหาคำตอบระหว่างทางหลายอย่าง กล่าวคือ ผู้เรียนจะต้องช่วยให้ออกด้วยตนเองว่า โจทย์ข้อนี้มีหลายวัตถุประสงค์ซ่อนอยู่ข้างใน นักเรียนนักศึกษาจำนวนมากจะทำข้อนี้ไม่ได้เว้นเสียแต่ว่าได้ทำข้อ 1 - 4 มาก่อน คนที่ได้ทำข้อ 5 นี้โดยที่ไม่เห็นข้อ 1 - 4 มาก่อนมักจะงงว่าควรจะคำนวณอะไรบ้าง)

6. จากข้อ 5 เด็กชายต้นจะต้องใช้บริการกี่ชั่วโมงต่อเดือนเป็นอย่างน้อย จึงจะทำให้การเป็นสมาชิกมีค่าใช้จ่ายต่อเดือนน้อยกว่าการไม่เป็นสมาชิก (จำนวนชั่วโมงต้องอยู่ในรูปจำนวนเต็ม)

Level 3 : แบบเชิงสัญลักษณ์ (Symbolic)

7. ร้านอินเทอร์เน็ตคาเฟ่ร้านหนึ่งคิดค่าใช้จ่ายค่าบริการ 15 บาทต่อชั่วโมง ทางร้านต้องการดึงดูดลูกค้าที่ใช้อินเทอร์เน็ตหรือเล่นเกมส์มาก จึงมีระบบสมาชิกร้านซึ่งผู้สมัครสมาชิกจะเสียค่าสมาชิก 100 บาทต่อเดือนและจะได้ส่วนลด $X\%$ จากค่าบริการ โดยที่ X มีค่ามากกว่า 0 และ ไม่เกิน 100 จงคำนวณค่าใช้จ่ายทั้งหมดของผู้ที่เป็นสมาชิก และของผู้ที่ไม่เป็นสมาชิกเมื่อมีการใช้บริการ Y ชั่วโมงต่อเดือน (คำตอบต้องอยู่ในรูปของค่า X และ Y)

8. จากข้อ 7 จำนวนชั่วโมง Y จะต้องมามีค่าไม่น้อยกว่าเท่าใดจึงจะคุ้มค่าที่สมัครสมาชิก (คำตอบต้องอยู่ในรูปของค่า X) (เนื่องจากจำนวนชั่วโมงต้องอยู่ในรูปจำนวนเต็ม คำตอบต้องระบุให้มีการปัดเศษขึ้นหรือลงอย่างชัดเจนด้วย)

Level 4 : แบบขั้นสูงสุด (Ultimate)

9. ร้านอินเทอร์เน็ตคาเฟ่ร้านหนึ่งคิดค่าใช้จ่ายค่าบริการ M บาทต่อชั่วโมง ทางร้านต้องการดึงดูดลูกค้าที่ใช้อินเทอร์เน็ตหรือเล่นเกมส์มาก จึงมีระบบสมาชิกร้านซึ่งผู้สมัครสมาชิกจะเสียค่าสมาชิก N บาทต่อเดือนและจะได้ส่วนลด $X\%$ จากค่าบริการ โดยที่ X มีค่ามากกว่า 0 และ ไม่เกิน 100 จงคำนวณค่าใช้จ่ายทั้งหมดของผู้ที่เป็นสมาชิก และของผู้ที่ไม่เป็นสมาชิกเมื่อมีการใช้บริการ Y ชั่วโมงต่อเดือน และหาด้วยว่าจำนวนชั่วโมงต่อเดือนจะต้องมีค่าไม่น้อยกว่าเท่าใดจึงจะคุ้มค่าที่จะสมัครสมาชิก

ปัญหา ขับรถตามให้ทัน

(แนะนำให้ทำข้อ 1, 3, 5, 6 และ 8)

Level 1 : แบบพื้นฐาน (Basic)

1. นาย A ขับรถตามรถของนาย B ซึ่งปัจจุบันนำหน้ารถนาย A อยู่ 5 กิโลเมตร หากนาย A และนาย B ขับรถด้วยความเร็ว 100 และ 80 กิโลเมตรต่อชั่วโมง จงคำนวณว่าจะต้องใช้เวลากี่นาทีที่นาย A จึงจะตามนาย B ทัน
2. นาย A ขับรถตามรถของนาย B ซึ่งปัจจุบันนำหน้ารถนาย A อยู่ 5 กิโลเมตร หากนาย A และนาย B ขับรถด้วยความเร็ว 100 และ 80 กิโลเมตรต่อชั่วโมง จงคำนวณว่านาย A จะต้องขับรถไปอีกกี่กิโลเมตรจึงจะตามทัน B
3. นาย A ขับรถตามรถของนาย B ซึ่งปัจจุบันนำหน้ารถนาย A อยู่ 5 กิโลเมตร หากนาย B ขับรถด้วยความเร็ว 80 กิโลเมตรต่อชั่วโมง จงคำนวณว่านาย A จะต้องขับรถด้วยความเร็วเท่าใดจึงจะตามนาย B ทันในเวลา 10 นาที
4. นาย A ขับรถตามรถของนาย B ซึ่งปัจจุบันนำหน้ารถนาย A อยู่ 5 กิโลเมตร หากนาย B ขับรถด้วยความเร็ว 80 กิโลเมตรต่อชั่วโมง จงคำนวณว่านาย A จะต้องขับรถด้วยความเร็วเท่าใดจึงจะตามนาย B ทันในระยะทาง 15 กิโลเมตรจากจุดที่นาย A อยู่ ณ ปัจจุบัน

Level 2 : แบบหลายจุดประสงค์ (Multi-Objective)

5. ตำรวจ A กำลังขับรถตำรวจไล่ตามรถของผู้ร้าย B แต่เนื่องจากอำนาจการจับกุมของตำรวจถูกจำกัดอยู่เฉพาะภายในท้องที่ของตน ตำรวจ A จึงต้องเร่งเครื่องตามรถผู้ร้ายให้ทัน ก่อนที่ผู้ร้ายจะหลบหนีออกนอกท้องที่ได้ ตำรวจจึงมีการคำนวณความเร็วที่ต้องใช้เพื่อตรวจดูว่าจะขับตามได้จริงและปลอดภัยเพียงพอหรือไม่ ถ้าหากผู้ร้ายต้องขับรถต่ออีกเพียง 4 กิโลเมตรก็จะพ้นท้องที่ของตำรวจ A และรถตำรวจตามหลังอยู่ 2 กิโลเมตร และหากผู้ร้ายขับรถด้วยความเร็ว 120 กิโลเมตรต่อชั่วโมง ตำรวจต้องขับรถด้วยความเร็วเท่าใดจึงจะตามรถผู้ร้ายทันก่อนที่ผู้ร้ายจะขับพ้นเขตท้องที่ทำการของตำรวจ A ไปได้ กำหนดให้รถตำรวจสามารถขับได้อย่างปลอดภัยถ้าความเร็วไม่เกิน 180 กิโลเมตรต่อชั่วโมง จงตอบว่ารถตำรวจนี้สามารถขับตามผู้ร้ายได้ทันตามมาตรฐานความปลอดภัยหรือไม่

Level 3 : แบบเชิงสัญลักษณ์ (Symbolic)

6. นาย A ขับรถตามรถของนาย B ซึ่งปัจจุบันนำหน้ารถนาย A อยู่ D กิโลเมตร โดยที่ $D > 0$ หากนาย A และนาย B ขับรถด้วยความเร็ว 100 และ 80 กิโลเมตรต่อชั่วโมง จงคำนวณว่านาย A จะต้องขับรถไปอีกกี่กิโลเมตรจึงจะตามทัน B (คำตอบต้องอยู่ในรูปของตัวแปร D)
7. นาย A ขับรถตามรถของนาย B ซึ่งปัจจุบันนำหน้ารถนาย A อยู่ 5 กิโลเมตร หากนาย A และนาย B ขับรถด้วยความเร็ว X และ Y กิโลเมตรต่อชั่วโมง โดยที่ $X, Y > 0$ จงคำนวณว่าต้องใช้เวลากี่นาทีที่นาย A จึงจะตามนาย B ทัน จงบอกเงื่อนไขที่เพียงพอของค่า X และ Y ด้วยว่าค่าของ X และ Y จะต้องมีความสัมพันธ์อย่างไร นาย A จึงจะสามารถขับรถตามนาย B ได้ทัน (คำตอบต้องอยู่ในรูปของตัวแปร X และ Y)

Level 4 : แบบขั้นสูงสุด (Ultimate)

8. นาย A ขับรถตามรถของนาย B ซึ่งปัจจุบันนำหน้ารถนาย A อยู่ D กิโลเมตร หากนาย A และนาย B ขับรถด้วยความเร็ว X และ Y กิโลเมตรต่อชั่วโมง จงคำนวณว่าจะต้องใช้เวลากี่นาทีที่นาย A จึงจะตามนาย Bทัน และหากน้ำมันหนึ่งลิตรสามารถขับรถได้เป็นระยะทาง M กิโลเมตร จงคำนวณเพิ่มเติมด้วยว่าทั้ง A และ B จะต้องใช้น้ำมันรวมกันกี่ลิตรนับจากปัจจุบันจนถึงขณะที่นาย A ขับทันนาย Bพอดี (คำตอบต้องอยู่ในรูปของตัวแปร D, X, Y และ M)

9. ตำรวจกำลังขับรถตำรวจไล่ตามรถของผู้ร้าย แต่เนื่องจากอำนาจการจับกุมของตำรวจถูกจำกัดอยู่เฉพาะภายในท้องที่ของตน ตำรวจจึงต้องเร่งเครื่องตามรถผู้ร้ายให้ทัน ก่อนที่ผู้ร้ายจะหลบหนีออกนอกท้องที่ได้ ตำรวจจึงมีการคำนวณความเร็วที่ต้องใช้เพื่อตรวจดูว่าจะขับตามได้จริงและปลอดภัยเพียงพอหรือไม่ ถ้าหากผู้ร้ายต้องขับรถต่ออีกเพียง S กิโลเมตรก็จะพ้นท้องที่ของตำรวจและขณะนี้รถตำรวจตามหลังอยู่ D กิโลเมตร หากผู้ร้ายขับรถด้วยความเร็ว Y กิโลเมตรต่อชั่วโมง ตำรวจต้องขับรถด้วยความเร็วเท่าใดจึงจะตามรถผู้ร้ายทันก่อนที่ผู้ร้ายจะขับพ้นเขตท้องที่ทำการของตำรวจไปได้ กำหนดให้รถตำรวจสามารถขับได้อย่างปลอดภัยถ้าความเร็วไม่เกิน 180 กิโลเมตรต่อชั่วโมง จงตอบว่ารถตำรวจนี้สามารถขับตามผู้ร้ายได้ทันตามมาตรฐานความปลอดภัยหรือไม่ ถ้าได้ต้องขับที่ความเร็วอย่างน้อยเท่าใด (คำตอบต้องอยู่ในรูปของตัวแปร $D, S, X,$ และ Y)

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่หนึ่ง ชุดที่สอง

พื้นฐานการวิเคราะห์และแก้ปัญหา

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

บทนำ

ในแบบฝึกหัดชุดที่แล้ว เราได้ลองทำการคำนวณกับปัญหาที่ข้อมูลเข้ามีชุดเดียว และได้คำตอบสุดท้ายออกมาจากข้อมูลเข้านั้นโดยตรง แต่ปัญหาจำนวนมากข้อมูลเข้ามีอยู่หลายชุดและข้อมูลเข้าชุดก่อนหน้ามีผลต่อผลลัพธ์ของข้อมูลเข้าในชุดต่อมา

เช่น ถ้านาย A และ B ต้องการยืมรถคันเดียวกัน โดยนาย A ต้องการยืม 3 วันและนาย B ต้องการยืม 2 วัน หากเราต้องเขียนโปรแกรมเพื่อหาว่านาย A และ นาย B จะได้รถวันใด เราจะพบว่าลำดับของการยืมมีผลต่อคำตอบ กล่าวคือ หากนาย A ยื่นเรื่องขอยืมก่อนนาย B นาย A จะได้รถวันแรก ส่วนนาย B จะได้รถในวันที่ 4 ในทางกลับกัน หากนาย B ยื่นเรื่องก่อน นาย B ก็จะได้รถวันแรก ส่วนนาย A ได้รถวันที่ 3 เป็นต้น

ปัญหาจำนวนมากมีลักษณะและความสัมพันธ์ของข้อมูลในรูปแบบดังกล่าว คือข้อมูลเข้ามีหลายชุดและแต่ละชุดส่งผลต่อเนื่องกันไป ในแบบฝึกหัดนี้มีจุดประสงค์เพื่อให้ผู้เรียนได้ฝึกฝนการแก้ปัญหาในลักษณะดังกล่าว พร้อมทั้งแสดงให้เห็นว่า 'ปัญหาแต่ละอัน อาจจะมีวิธีคำนวณหลายแบบ แต่ละแบบก็มีความยากง่ายและความสะดวกที่แตกต่างกันไป'

ปัญหา จองรถตู้ [reserve_van_shift]

บริษัทแห่งหนึ่งมีรถตู้อยู่ 3 คันที่พนักงานสามารถนำไปใช้ได้ รถตู้คันที่หนึ่ง สอง และ สาม มีรหัสประจำรถว่า A,B และ C ตามลำดับ ข้อกำหนดในการนำไปใช้มีอยู่ว่าพนักงานจะต้องทำการจองรถก่อน โดยคำสั่งจองจะต้องระบุจำนวนวันที่จะใช้ จากนั้นผู้จองจะได้รถตู้ที่ว่างให้ใช้เร็วที่สุดเท่าที่หาได้จากหนึ่งในสามคันนั้น

ในกรณีที่รถตู้ว่างให้ใช้เร็วที่สุดมากกว่าหนึ่งคันและ A ว่างให้ใช้เร็วที่สุด A จะถูกเลือกก่อน B และ C (เป็นไปได้ว่าจะว่างให้ใช้เร็วที่สุดพร้อมกันทั้งสามคัน หรือแค่สองคันซึ่งเป็น A กับ B หรือ A กับ C ก็ได้) ถ้า A ไม่ได้ว่างให้ใช้เร็วที่สุด แต่เป็น B กับ C ที่ว่างให้ใช้ได้เร็วที่สุดพร้อมกันทั้งคู่ รถ B จะถูกเลือกก่อน C นอกจากนี้การจองจะให้ความสำคัญกับคำสั่งจองที่มาก่อนเสมอ สำหรับการจองแต่ละครั้ง ผู้จองจะได้รับคำตอบกลับมามีว่าจะได้ใช้รถคันใด ซึ่งมีเกณฑ์การเลือกรถเป็นไปตามที่อธิบายไว้ก่อนหน้า

จงหาว่าจากจำนวนคำสั่งจอง N คำสั่งพร้อมทั้งรายละเอียดเกี่ยวกับคำสั่งจองที่กำหนดให้ รถคันใดจะถูกนำไปใช้กับคำสั่งจองแต่ละคำสั่งโดยหากเป็นรถ A ก็ให้เขียนข้อความว่า A และขึ้นบรรทัดใหม่ ถ้าเป็นรถ B หรือ C ก็เขียนผลลัพธ์ออกมาในลักษณะเดียวกัน [มีตัวอย่างและคำอธิบายเพิ่มเติมท้ายโจทย์] กำหนดเพิ่มเติมว่าในตอนแรกรถตู้ทั้งสามคันว่างและพร้อมใช้ทั้งหมด ส่วนคำสั่งจองทุกคำสั่งได้มาก่อนเปิดให้บริการรถตู้ตั้งแต่วันแรก

ข้อมูลเข้า

1. บรรทัดแรกระบุจำนวนคำสั่งจองเป็นจำนวนเต็มบวก N โดยที่ $N \geq 1$
2. บรรทัดที่ 2 ถึง $N+1$ ระบุคำสั่งจองเรียงตามลำดับการขอ (บรรทัดที่มาก่อนหมายถึงการจองก่อน) ในแต่ละบรรทัดประกอบด้วยเลขจำนวนเต็มบวกหนึ่งตัวคือ t โดยที่ $1 \leq t \leq 15$ (นั่นคือจองรถตู้ได้ครั้งละ 1 ถึง 15 วัน)

ผลลัพธ์

มีทั้งหมด N บรรทัด โดยแต่ละบรรทัดระบุว่ารหัสรถคันใดจะถูกนำไปใช้กับคำสั่งจองแต่ละคำสั่ง โดยผลลัพธ์เรียงตามลำดับคำสั่งจอง

ตัวอย่าง

ตัวอย่างที่หนึ่ง		ตัวอย่างที่สอง	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
6		6	
3	A	1	A
1	B	2	B
2	C	2	C
2	B	1	A
2	C	1	A
1	A	3	B

อธิบายตัวอย่างที่หนึ่ง

บรรทัดแรก: เลข 6 ในคือจำนวนคำสั่งที่ต้องคำนวณ

บรรทัดที่สอง: เลข 3 คือจำนวนวันที่จะใช้รถ เนื่องจากในตอนแรกรถทุกคันว่างหมดจึงเลือกใช้รถ A ตรงนี้ควรจำไว้ด้วยว่ารถ A จะว่างใช้อีกทีในวันที่ 4

บรรทัดที่สาม: เลข 1 คือจำนวนวันที่จะใช้รถ เนื่องจากตอนนี้รถที่ว่างใช้ได้เร็วที่สุดคือ B และ C จึงเลือก B ก่อน เช่นเดิมจำไว้ด้วยว่า B จะว่างใช้อีกทีในวันที่ 2

บรรทัดที่สี่: เลข 2 คือจำนวนวันที่จะใช้รถ เนื่องจากตอนนี้รถที่ว่างใช้ได้เร็วที่สุดคือ C จึงเลือก C เช่นเดิมจำไว้ด้วยว่า C จะว่างใช้อีกทีในวันที่ 3

บรรทัดที่ห้า: เลข 2 คือจำนวนวันที่จะใช้รถ เนื่องจากตอนนี้รถที่ว่างใช้ได้เร็วที่สุดคือ B จึงเลือก B เช่นเดิมจำไว้ด้วยว่า B จะว่างใช้อีกทีในวันที่ $2 + 2 = 4$

บรรทัดที่หก: เลข 2 คือจำนวนวันที่จะใช้รถ เนื่องจากตอนนี้รถที่ว่างใช้ได้เร็วที่สุดคือ C จึงเลือก C เช่นเดิมจำไว้ด้วยว่า C จะว่างใช้อีกทีในวันที่ $3 + 2 = 5$

บรรทัดที่เจ็ด: เลข 1 คือจำนวนวันที่จะใช้รถ เนื่องจากตอนนี้รถที่ว่างใช้ได้เร็วที่สุดคือ A และ B จึงเลือก A

คำถามของเราและคำตอบของคุณ

จงระบุวารรถู้คันใดที่จะถูกนำไปใช้ โดยเติมคำตอบลงในที่ว่างที่เตรียมไว้ให้ในช่องผลลัพธ์ ทั้งนี้คำตอบบางส่วนได้ถูกเติมไว้ให้เรียบร้อยแล้ว ผู้เข้าสอบจะต้องเติมส่วนที่เหลือให้ครบ

สถานการณ์ที่หนึ่ง		สถานการณ์ที่สอง		สถานการณ์ที่สาม	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
7		10		15	
2	A	5	A	2	A
2	B	2	B	3	B
1	C	4	C	2	C
1		7		2	
1		3		3	
3		2		2	
1		1		2	
		5		3	
		1		2	
		1		2	
				3	
				2	
				2	
				3	
				2	

ปัญหา วันที่ตัดเสื้อแล้วเสร็จ [due_date]

โรงงานตัดเย็บเสื้อแห่งหนึ่ง รับคำสั่งตัดเสื้อจากลูกค้ามาทั้งหมด N คำสั่ง โดยในแต่ละคำสั่งจะเป็นปริมาณเสื้อที่ลูกค้าต้องการ กำหนดให้โรงงานมีพนักงานทั้งหมด K คนและคนงานหนึ่งคนจะผลิตเสื้อได้ 100 ตัวต่อวัน นอกจากนี้ โรงงานจะทำการผลิตเสื้อจากคำสั่งตัดเสื้อคำสั่งแรกเสร็จแล้วจึงทำคำสั่งตัดที่สอง สาม เรียงตามลำดับเช่นนี้ไปตลอด เนื่องจากการคำนวณเวลาที่งานจะแล้วเสร็จเป็นสิ่งสำคัญในการประมาณวันรับสินค้าของลูกค้า โรงงานจึงพยายามทำการคำนวณเวลาที่ตัดเสื้อของแต่ละคำสั่งแล้วเสร็จ

จงหาว่าจากข้อมูลคำสั่งตัดเสื้อที่กำหนดให้ โรงงานจะตัดเย็บเสื้อผ้าของแต่ละคำสั่งเสร็จในวันที่เท่าใดนับจากวันแรก (วันแรกคือวันที่หนึ่ง)

หมายเหตุ ถ้าคนงานตัดเสื้อของคำสั่งซื้อคำสั่งที่หนึ่งเสร็จ และยังมีเหลือเวลาอีกก็จะทำการตัดเสื้อของคำสั่งตัดเสื้อที่ตามมาเรื่อย ๆ จนกว่าคนงานคนดังกล่าวจะตัดเสื้อครบร้อยตัวในวันนั้น หรือคำสั่งตัดเสื้อทุกอันถูกทำงานเสร็จทุกงานแล้ว เช่น หากคนงาน 2 คน และมี 4 คำสั่งตัดเสื้อจากลูกค้า เรียงตามลำดับดังนี้

90

60

70

40

ความสามารถในการตัดเสื้อต่อวันของโรงงานคือ $100 \times 2 = 200$ ตัวต่อวัน (คนงานสองคน) หลังจากทำคำสั่งแรกเสร็จซึ่งคิดรวมเป็นเสื้อได้ 90 ตัว คนงานจะทำคำสั่งที่สองต่อไปในวันเดียวกัน ซึ่งทำให้ผลิตเสื้อออกมารวมได้ 150 ตัว จากสองคำสั่งซื้อ คนงานจึงตัดเสื้อตามคำสั่งตัดที่สามต่อ แต่จะตัดได้แค่ 50 ตัว ที่เหลืออีก 20 ตัวในคำสั่งตัดที่สามจะต้องมาทำต่อในวันถัดไป พอถึงวันถัดมาคนงานก็ตัดเย็บเสื้อ 20 ตัวนั้นจนเสร็จและตัดเสื้อตามคำสั่งซื้อสุดท้ายเสร็จในวันเดียวกัน

ข้อมูลเข้า

1. บรรทัดแรกระบุจำนวนคนงาน (ค่า K) เป็นจำนวนเต็มบวก
2. บรรทัดที่สองระบุจำนวนคำสั่งตัดเสื้อ (ค่า N) เป็นจำนวนเต็มบวก
3. อีก N บรรทัดต่อมา ระบุจำนวนเสื้อที่สั่งตัดในแต่ละคำสั่งตัด หนึ่งคำสั่งต่อหนึ่งบรรทัด เรียงจากคำสั่งแรกไปคำสั่งสุดท้าย

ผลลัพธ์

มี N บรรทัดเรียงตามลำดับคำสั่งตัดเสื้อคือ ในแต่ละบรรทัดระบุว่าคำสั่งตัดเสื้อแต่ละคำสั่งจะเสร็จในวันที่เท่าใด (ให้นับจำนวนวันเริ่มจาก 1 ไปเรื่อย ๆ) บรรทัดแรกแทนวันที่ตัดเสื้อของคำสั่งซื้อแรกแล้วเสร็จ

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
2		2		5	
4		4		10	
90	1	210	2	400	1
60	1	200	3	400	2
70	2	190	3	500	3
40	2	200	4	1700	6
				5000	16
				1300	19
				750	21
				280	21
				375	22
				2000	26

คำถามของเราและคำตอบของคุณ

จงระบุจากข้อมูลคำสั่งตัดเสื้อที่กำหนดให้ โรงงานจะตัดเย็บเสื้อผ้าของแต่ละคำสั่งเสร็จในวันที่เท่าใด

สถานการณ์ที่หนึ่ง		สถานการณ์ที่สอง		สถานการณ์ที่สาม	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
3		2		5	
9		9		15	
290	1	290	2	1700	4
160		160		5000	
700		700		1300	
40		40		750	
50		50		280	
30		30		375	
20		20		2025	
2000		2000		400	
1000		1000		400	
				500	
				3000	
				400	
				200	
				800	
				1700	

ปัญหา คัดเลือกนักร้อง [the_voice_su]

ในการประกวดร้องเพลงรายการ the_voice_su มีกรรมการอยู่สองคน คนแรกเป็นชายและคนที่สองเป็นหญิง กรรมการแต่ละคนจะทำการคัดเลือกผู้ที่เข้ารอบ โดยกระบวนการคัดเลือกเป็นดังนี้

1. ผู้เข้าประกวดจะมาร้องเพลงทีละคนเรียงจากหมายเลข 1 จนถึงหมายเลข N
2. กรรมการทั้งสองท่านจะให้คะแนนเป็นจำนวนเต็มระหว่าง 0 ถึง 10
3. กรรมการจะพยายามคัดเลือกผู้ประกวดให้เข้ามาอยู่ในความดูแลของตนเพื่อฝึกฝนและเข้าสู่รอบต่อไป นอกจากนี้กรรมการแต่ละคนจะดูแลผู้เข้าประกวดได้ไม่เกิน K คน ผู้ที่เข้ารอบในการประกวดคือผู้ที่ได้รับการดูแลจากกรรมการ
4. ในการให้คะแนน ถ้ากรรมการท่านใดให้คะแนน 9 หรือ 10 กับผู้เข้าประกวด จะหมายความว่าผู้เข้าประกวดคนดังกล่าวมีความสามารถในระดับสูง และกรรมการท่านนั้นต้องการรับมาดูแลด้วย อย่างไรก็ตามกรรมการท่านนั้นจะต้องมีนักร้องในความดูแลของตนยังไม่ครบ K คน ไม่เช่นนั้นก็จะรับผู้เข้าประกวดมาดูแลไม่ได้
5. ในบางครั้งกรรมการทั้งสองคนต่างเห็นว่าผู้เข้าประกวดมีความสามารถในระดับสูงและต้องการรับผู้ประกวดคนเดียวกันมาอยู่ในความดูแล ถ้ากรรมการทั้งสองยังมีคนไม่ครบ K คน ผู้เข้าประกวดจะเลือกกรรมการที่เป็นชายหรือหญิงเช่นเดียวกันตน แต่หากมีกรรมการเพียงหนึ่งท่านที่ยังรับผู้เข้าประกวดมาดูแลได้ ผู้เข้าประกวดก็จะอยู่ในความดูแลของกรรมการคนดังกล่าวโดยอัตโนมัติ ในกรณีที่เลวร้ายที่สุดก็คือกรรมการทั้งสองท่านไม่สามารถรับผู้เข้าประกวดมาดูแลเพิ่มได้ ผู้เข้าประกวดก็จะตกรอบ ไม่ว่าจะได้คะแนนดีสักเพียงใดก็ตาม
6. การคัดเลือกจะเรียงตามหมายเลขผู้เข้าประกวดตามที่ได้รับไว้ ทำให้ผู้ที่มีหมายเลขอันดับท้าย ๆ มีสิทธิ์น้อยลงที่จะได้รับการคัดเลือก เพราะกรรมการทั้งสองท่านอาจจะได้ผู้เข้าประกวดในความดูแลครบแล้วก็ได้ อย่างไรก็ตามทางผู้จัดการแข่งขันเห็นว่า อย่างน้อยควรให้ผู้เข้าประกวดทุกคนได้แสดงความสามารถให้เห็นแก่สาธารณชน ผู้เข้าประกวดทุกคนจึงจะร้องเพลงไปจนครบทุกคนและกรรมการก็จะยังทำการให้คะแนนกับผู้เข้าแข่งขันทุกคนตามปกติ

จงทำการคำนวณว่านักร้องหมายเลขใดบ้างที่เข้ารอบและผู้ที่ยังอยู่ในความดูแลของกรรมการท่านใด

ข้อมูลเข้า

1. บรรทัดแรกเป็นจำนวนเต็มบวก N และ K ตามลำดับ
2. บรรทัดที่ 2 ถึง N + 1 ระบุเพศและคะแนนของผู้เข้าประกวด หนึ่งผู้เข้าประกวดต่อหนึ่งบรรทัด เรียงจากผู้เข้าประกวดหมายเลขที่หนึ่งไปจนหมายเลขที่ N แต่ละบรรทัดมีตัวเลขจำนวนเต็มสามตัวซึ่งระบุเพศและข้อมูลคะแนนของผู้เข้าประกวด ตัวเลขแรกเป็นเพศ ซึ่ง 1 แทนเพศชายและ 2 แทนเพศหญิง ตัวเลขที่สองและสามในบรรทัดแทนคะแนนจากกรรมการคนที่หนึ่งและสองตามลำดับ

ผลลัพธ์

มีทั้งหมดไม่เกิน 2K บรรทัด โดยแต่ละบรรทัดระบุหมายเลขของผู้เข้าประกวดที่เข้ารอบ ตามด้วยกรรมการที่รับผู้เข้าประกวดไว้ในความดูแล โดยใช้เลข 1 แทนกรรมการคนแรกซึ่งเป็นชาย และเลข 2 แทนกรรมการคนที่สองซึ่งเป็นหญิง การเรียงลำดับการแสดงผลนั้น ให้เรียงตามหมายเลขของผู้เข้าประกวด **หมายเหตุ** เป็นไปได้ว่าจำนวนผู้เข้าประกวดที่เข้ารอบอาจจะมีน้อยกว่า 2K ถ้ากรรมการเห็นว่าผู้เข้าประกวดที่มีความสามารถในระดับสูงมีไม่มากนัก

ตัวอย่าง

ตัวอย่างที่หนึ่ง		ตัวอย่างที่สอง		ตัวอย่างที่สาม	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
6 2		7 2		7 3	
1 9 10	1 1	1 8 7		1 8 9	1 2
1 8 10	2 2	2 9 9	2 2	2 9 9	2 2
2 9 9	3 2	1 7 8		1 8 7	
2 8 9		1 10 8	4 1	1 7 8	
2 10 9	5 1	2 10 9	5 2	2 9 10	5 2
1 10 10		2 9 9	6 1	1 10 10	6 1
		1 9 9		2 8 9	

อธิบายตัวอย่างที่หนึ่ง

บรรทัดแรก: เลข 6 คือจำนวนผู้เข้าประกวด เลข 2 คือค่า K ซึ่งแสดงจำนวนผู้เข้าประกวดที่กรรมการสามารถรับมาดูแลเพื่อฝึกฝนและให้เข้ารอบต่อไปได้

บรรทัดที่สอง: เลข 1 แสดงว่าผู้เข้าประกวดเป็นชาย ส่วน 9 และ 10 คือคะแนนจากกรรมการคนที่หนึ่งและสองตามลำดับ จากคะแนนแสดงว่ากรรมการทั้งสองท่านต้องการรับผู้เข้าประกวดคนนี้ไว้ เนื่องจากมีที่ว่างพร้อมกัน ผู้เข้าประกวดจึงเลือกที่จะอยู่กับกรรมการคนแรกเพราะเป็นผู้ชายเหมือนกัน

บรรทัดที่สาม: เลข 1 แสดงว่าผู้เข้าประกวดเป็นชาย ส่วน 8 และ 10 คือคะแนนจากกรรมการคนที่หนึ่งและสองตามลำดับ จากคะแนนแสดงว่ามีเฉพาะกรรมการคนที่สองที่ต้องการรับผู้เข้าประกวดคนนี้ไว้ เนื่องจากกรรมการคนที่สองยังมีที่ว่าง ผู้เข้าประกวดหมายเลขสองจึงเข้ารอบและอยู่กับกรรมการคนที่สอง

บรรทัดที่สี่: เลข 2 แสดงว่าผู้เข้าประกวดเป็นหญิง ส่วน 9 และ 9 คือคะแนนจากกรรมการคนที่หนึ่งและสองตามลำดับ จากคะแนนแสดงว่ากรรมการทั้งสองท่านต้องการรับผู้เข้าประกวดคนนี้ไว้ และมีที่ว่างพร้อมกัน ผู้เข้าประกวดจึงเลือกที่จะอยู่กับกรรมการคนที่สองเพราะเป็นผู้หญิงเหมือนกัน

บรรทัดที่ห้า: เลข 2 แสดงว่าผู้เข้าประกวดเป็นหญิง ส่วน 8 และ 9 คือคะแนนจากกรรมการคนที่หนึ่งและสองตามลำดับ จากคะแนนแสดงว่ามีเฉพาะกรรมการคนที่สองที่ต้องการรับผู้เข้าประกวดคนนี้ไว้ แต่กรรมการที่สองยังไม่มีที่ว่างอีกแล้ว ผู้เข้าประกวดจึงตรอไปตามกติกา

บรรทัดที่หก: จากคะแนนแสดงว่ากรรมการทั้งสองท่านต้องการรับผู้เข้าประกวดคนนี้ไว้ แต่มีเฉพาะกรรมการคนแรกที่ยังมีที่ว่าง ผู้เข้าประกวดจึงอยู่ในความดูแลของกรรมการคนที่หนึ่ง

บรรทัดที่เจ็ด: จากคะแนนแสดงว่ากรรมการทั้งสองท่านต้องการรับผู้เข้าประกวดคนนี้ไว้ แต่ไม่มีใครมีที่ว่าง ผู้เข้าประกวดจึงตรอไปตามกติกาไม่ว่าจะได้คะแนนมากสักเพียงใดก็ตาม

หมายเหตุ โปรแกรมของเราไม่จำเป็นที่จะต้องรับอินพุตครบทุกบรรทัดก็ได้

คำถามของเราและคำตอบของคุณ

จรรยาบรรณจากข้อมูลเข้าที่ให้ไป ผู้เข้าประกวดรายใดที่เข้ารอบและอยู่กับกรรมการท่านใด การเรียงลำดับคำตอบจะเป็นไปตามหมายเลขผู้เข้าประกวด

สถานการณ์ที่หนึ่ง		สถานการณ์ที่สอง		สถานการณ์ที่สาม	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
10 3		15 5		20 7	
1 10 9	1 1	1 10 8	1 1	2 6 8	
2 10 9		2 10 5		1 5 7	
2 5 9		1 7 9		1 8 9	3 2
2 10 10		2 10 10		2 7 6	
1 8 6		2 5 9		1 9 5	
1 6 7		1 8 7		1 10 6	
1 5 8		1 9 9		2 6 6	
2 7 8		1 8 10		2 10 10	
2 9 8		1 9 6		2 10 7	
2 7 10		1 8 9		1 5 10	
		2 10 5		1 6 8	
		1 6 8		2 10 10	
		2 5 9		2 6 8	
		1 6 6		2 5 7	
		2 7 5		2 10 10	
				1 9 6	
				2 9 9	
				1 8 10	
				1 10 9	
				2 10 7	

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สอง

การวางแผนการคำนวณด้วยโฟลวชาร์ตและซูดโค้ด: การใช้โครงสร้างตัดสินใจ (IF)

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

คำสั่ง

จงเขียนโฟลวชาร์ตที่แสดงลำดับการทำงานในแบบฝึกหัดแต่ละข้อต่อไป ในบางข้อต้องเขียนซูดโค้ดหรือภาษาซีที่เทียบเท่ากันด้วย ทั้งนี้ขอให้ผู้เรียนเขียนโค้ดที่เทียบเท่ากับโฟลวชาร์ตที่ได้ด้วย ไม่เช่นนั้นจะถือว่าผิดในแบบฝึกหัดนี้

แบบฝึกหัดชุดที่ 1: การรับข้อมูลเข้า การคำนวณ และการแสดงผลลัพธ์

คำชี้แจง แบบฝึกหัดชุดนี้ไม่ต้องใช้โครงสร้างตัดสินใจหรือแยกกรณี IF-ELSE ใดๆก็ตาม โจทย์ถามหาของมากกว่าหนึ่งอย่าง ทำให้เราต้องคำนวณคำตอบทุกอย่างให้ครบถ้วน ซึ่งผู้เรียนที่ยังอ่อนประสบการณ์มักจะพลาดที่จุดนี้

1. จงเขียนโฟลวชาร์ตที่สอดคล้องกับสูตรที่เป็นคำตอบของเราในข้อ 7 จากชุดปัญหา*สมัครสมาชิก* ในแบบฝึกหัดของสัปดาห์แรก ชุดที่หนึ่ง โดยให้ X และ Y เป็นข้อมูลเข้า
2. จงเขียนซูดโค้ดที่สอดคล้องกับโฟลวชาร์ตที่ได้ในข้อ 1

หมายเหตุ สำหรับแบบฝึกหัดในสัปดาห์ที่สองและสามนี้คนที่เรียนมาแล้วจะเขียนเป็นโค้ดภาษาซีแทนซูดโค้ดก็ได้

แบบฝึกหัดชุดที่ 2: การแยกกรณีการทำงานด้วยเงื่อนไข (condition) แบบ IF แบบพื้นฐาน

คำชี้แจง การใช้ IF แบบพื้นฐานจะมีลักษณะของการเลือกทำหรือไม่ทำ เช่น เราอาจจะตั้งเงื่อนไขว่า “ถ้าอากาศดี จะออกไปดูหนัง” นั่นคือมีความเป็นไปได้สองแบบดังนี้ (1) เราเลือกไปดูหนัง และ (2) ไม่ทำอะไรเลย เราจะลองฝึกใช้แนวคิดนี้กับการคำนวณทางคณิตศาสตร์ในโจทย์ข้างล่างนี้

3. จงเขียนโฟลวชาร์ตของโปรแกรมที่รับตัวเลขจำนวนเต็มมาหนึ่งจำนวน หากตัวเลขดังกล่าวหารด้วย 3 ลงตัว โปรแกรมจะพิมพ์ข้อความว่า '3' ออกมา ไม่เช่นนั้นโปรแกรมจะไม่พิมพ์อะไรออกมาเลย
4. จงเขียนซูดโค้ดที่สอดคล้องกับโฟลวชาร์ตที่ได้ในข้อ 3

แบบฝึกหัดชุดที่ 3: การแยกกรณีการทำงานด้วยเงื่อนไข (condition) แบบ IF-ELSE

คำชี้แจง แบบฝึกหัดชุดนี้ต้องการฝึกผู้เรียนให้ใช้โครงสร้างตัดสินใจและแยกกรณี IF-ELSE ซึ่งเป็นการเลือกทำทางใดทางหนึ่ง แต่ไม่ใช่ทั้งสองทาง เช่น หากเราตั้งเงื่อนไขว่า 'ถ้าอากาศดี จะออกไปดูหนัง ไม่เช่นนั้นจะเล่นเกมอยู่ที่บ้าน' นั่นคือเราไม่ได้ทำของทั้งสองอย่างพร้อมกัน แต่จะเลือกทำเพียงอย่างเดียว ขอให้สังเกตด้วยว่าเราอ้างอิงเงื่อนไขเพียงอย่างเดียวเพื่อเลือกทำกิจกรรมอันใดอันหนึ่ง เราจะลองฝึกใช้แนวคิดนี้กับการคำนวณในโจทย์ข้างล่างนี้

5. จงเขียนโฟลวชาร์ตของโปรแกรมที่รับตัวเลขจำนวนเต็มมาหนึ่งจำนวน (สมมติว่าตัวเลขดังกล่าวคือ X) หากตัวเลขดังกล่าวหารด้วย 3 ลงตัว โปรแกรมจะพิมพ์ข้อความว่า '3' ออกมา ไม่เช่นนั้นโปรแกรมจะพิมพ์คำว่า 'Not 3' ออกมาแทน

6. จงเขียนชุดโค๊ดที่สอดคล้องกับโฟลวชาร์ตที่ได้ในข้อ 5

=====

7. จงเขียนโฟลวชาร์ตที่สอดคล้องกับสูตรที่เป็นคำตอบของเราในข้อ 20 จากชุดปัญหา*ลางกระป๋อง* โดยให้ตัวแปรทุกตัวเป็นข้อมูลเข้า

8. จงเขียนชุดโค๊ดที่สอดคล้องกับโฟลวชาร์ตที่ได้ในข้อ 7

แบบฝึกหัดชุดที่ 4: การแยกกรณีการทำงานด้วยเงื่อนไข (condition) แบบ IF และ ELSE IF

คำชี้แจง นอกจากเราจะใช้ IF เพื่อตัดสินใจเกี่ยวกับสิ่งที่จะทำ เรายังใช้ IF เพื่อจำแนกประเภทหรือสถานะ ซึ่งหากประเภทที่ต้องการจำแนกมีมากกว่าสองอย่าง (เช่น การตัดเกรด) การใช้ IF และ ELSE IF ต่อกันไปมักจะเป็นทางเลือกที่เหมาะสมเพราะเราสามารถที่จะลดทอนการเขียนเงื่อนไขได้ ยกตัวอย่างเช่นการตัดเกรด โดยมีเกณฑ์ว่า ถ้าได้น้อยกว่า 40 คือตก, ได้ถึง 40 แต่น้อยกว่า 80 คือผ่าน, และได้ถึง 80 ขึ้นไปคือยอดเยี่ยม ขอให้ลองสังเกตชุดโค๊ดข้างล่างนี้

```
IF SCORE < 40 THEN
    PRINT 'FAIL'
ELSE IF SCORE >= 40 AND SCORE < 80 THEN
    PRINT 'PASS'
ELSE IF SCORE >= 80 THEN
    PRINT 'EXCELLENT'
END IF
```

จากการสังเกต เราพบว่าเงื่อนไขในการจำแนกประเภทเหล่านี้ถูกต้องสมบูรณ์ดี แต่เราก็สามารถละเว้นเงื่อนไขบางอย่างได้ เป็นต้นว่า ในเมื่อเราทราบว่าเกรดถ้าไม่เป็น FAIL หรือ PASS แล้วมันก็ต้องเป็น EXCELLENT ดังนั้นการจำแนกประเภทครั้งสุดท้ายจึงไม่จำเป็นต้องอาศัยเงื่อนไขใด ๆ เพิ่มเติม และเราสามารถเขียนใหม่ได้เป็น

```
IF SCORE < 40 THEN
    PRINT 'FAIL'
ELSE IF SCORE >= 40 AND SCORE < 80 THEN
    PRINT 'PASS'
ELSE
    PRINT 'EXCELLENT'
END IF
```

ด้วยข้อสังเกตทำนองเดียวกัน เราสามารถที่จะตัดทอนเงื่อนไขตรง ELSE IF ที่ว่า SCORE >= 40 AND SCORE < 80 ได้ด้วย เพราะการที่จะผ่านเงื่อนไข SCORE < 40 มาถึงเงื่อนไขตรง ELSE IF นี้ได้ ก็แสดงว่า SCORE < 40 ต้องเป็นเท็จ และข้อความตรงข้ามของมันซึ่งก็คือ SCORE >= 40 เป็นจริงโดยปริยาย

เมื่อเราทราบว่าข้อความ SCORE >= 40 เป็นจริงเมื่อมาถึงตำแหน่ง ELSE IF ดังกล่าว เราจึงไม่มีความจำเป็นที่จะต้องตรวจซ้ำ แต่ตัดออกไปได้ ทำให้เราสามารถลดทอนชุดโค๊ดได้เป็น

```
IF SCORE < 40 THEN
    PRINT 'FAIL'
```

```
ELSE IF SCORE < 80 THEN
    PRINT 'PASS'
ELSE
    PRINT 'EXCELLENT'
END IF
```

9. **ปัญหาคำนวณภาษี** สมมติว่าภาษีเงินได้ของบุคคลทั่วไปถูกแบ่งออกเป็นสี่กลุ่มคือ (1) กลุ่มที่ไม่ต้องเสียภาษี, (2) กลุ่มที่เสียภาษี 7%, (3) กลุ่มที่เสียภาษี 20% และ (4) กลุ่มที่เสียภาษี 30% โดยเกณฑ์การคิดภาษีจะคิดจากรายได้สุทธิ คือรายได้หลังหักรายการที่ใช้ลดหย่อนภาษีได้ ยกตัวอย่างเช่น นายศไกรมีรายได้ 300,000 บาทต่อปี และได้บริจาคเงินให้กับสภาอากาศไทยไป 50,000 บาท ซึ่งเป็นรายการที่ลดหย่อนภาษีได้ ในกรณีเช่นนี้รายได้สุทธิของนายศไกรก็คือ 250,000 บาท เมื่อได้รายได้สุทธิตามแล้วการแบ่งกลุ่มภาษียังจะถูกดำเนินการต่อไปตามเกณฑ์ต่อไปนี้

- (1) ถ้ารายได้สุทธิน้อยกว่า 200,000 บาทต่อปี ไม่ต้องเสียภาษีเงินได้
- (2) ถ้ารายได้ถึง 200,000 บาทต่อปี แต่น้อยกว่า 500,000 บาท จะเสียภาษีเงินได้ 7%
- (3) ถ้ารายได้ถึง 500,000 บาทต่อปี แต่น้อยกว่า 1,000,000 บาท จะเสียภาษีเงินได้ 20%
- (4) ถ้ารายได้ถึง 1,000,000 บาทต่อปี จะเสียภาษีเงินได้ 30%

จงเขียนซูโดโค้ดสำหรับการคำนวณและแสดงภาษีที่ต้องจ่าย เมื่อกำหนดเงินรายได้ S และยอดเงินรวมจากรายการที่ลดหย่อนภาษีได้ D มาให้ [หมายเหตุ วิธีคำนวณภาษีจริงซับซ้อนกว่านี้มาก ปัญหานี้เป็นเพียงรูปแบบที่ทำให้ง่ายเพื่อศึกษาเรียนรู้การใช้โครงสร้าง IF-ELSE-IF]

แบบฝึกหัดชุดที่ 5: ความเทียบเท่ากันของการใช้เงื่อนไขแบบต่าง ๆ

คำชี้แจง ในแบบฝึกหัดชุดที่ 3 และ 4 เราได้ใช้งานโครงสร้าง IF – ELSE และ IF – ELSE IF ซึ่งการใช้งานที่เราทำอยู่ในแบบฝึกหัดนั้นว่าเป็นตัวเลือกที่ดี แต่เราก็ได้เห็นในแบบฝึกหัดชุดที่ 4 แล้วว่าการเขียนเงื่อนไขในโครงสร้างนั้นสามารถทำได้หลายรูปแบบ และรูปแบบหนึ่งที่เราจะกล่าวถึงในแบบฝึกหัดชุดที่ 5 ก็คือการแยก IF – ELSE หรือ IF – ELSE IF ออกจากกันเป็น IF ที่อิสระจากกัน

พิจารณาแบบฝึกหัดข้อ 5 ที่ต้องการพิมพ์คำว่า '3' หรือ 'Not 3' ออกมา โดยวิธีการที่เราใช้ก่อนหน้านี้ เราเขียนซูโดโค้ดได้เป็น

```
IF X MOD 3 == 0 THEN
    PRINT 3
ELSE
    PRINT 'Not 3'
END IF
```

แต่ที่จริงเราสามารถแยกออกเป็นสองส่วนได้เป็น


```

IF X MOD 3 == 0 THEN
  PRINT 3
END IF
IF X MOD 3 != 0 THEN
  PRINT 'Not 3'
END IF

```

จากตัวอย่างที่ยกมา จะเห็นได้ว่าโครงสร้างการตัดสินใจสามารถถูกแยกออกเป็นหลายส่วนได้ และได้โค้ดที่เทียบเท่ากัน แต่การแยกออกเป็นหลายส่วน ในบางกรณีก็ทำให้เราพลาดการใช้ประโยชน์จากการลดทอนเงื่อนไขที่เราเคยทำได้ ในการเขียนแบบรวมต่อเนื่องกันไป เพื่อให้เห็นความเทียบเท่ากันดังกล่าว และทราบถึงข้อดีข้อเสียในแต่ละทางเลือก พร้อมทั้งฝึกทักษะในการใช้เงื่อนไขเพื่อแยกประเภท ผู้เรียนจึงควรลองทำแบบฝึกหัดข้างล่างนี้ด้วยตนเอง

10. จงเขียนชุดโค้ดโดยใช้วิธีแยก IF ออกเป็นหลายส่วนกับปัญหาคำนวณภาษี

=====

คำชี้แจง นอกจากการแยก IF ออกเป็นหลายส่วนแล้ว การใช้ IF – ELSE ก็ยังมีหลายรูปแบบได้ ขึ้นอยู่กับว่าเราจะเอาอะไรมาคิดก่อน เช่น ถ้าหากเราสนใจที่จะคิดกรณีที่หารไม่ลงตัวก่อน เราก็จะได้ชุดโค้ดเป็น

```

IF X MOD 3 != 0 THEN
  PRINT 'Not 3'
ELSE
  PRINT 3
END IF

```

การเลือกเงื่อนไขว่าจะเอาสิ่งใดขึ้นก่อนหลัง มีผลต่อความยากง่ายของโค้ด และอาจมีผลต่อความเร็วของโปรแกรมด้วย อย่างไรก็ตาม ในแบบฝึกหัดนี้ไม่ได้ต้องการเน้นทั้งสองประเด็น แต่ต้องการเน้นฝึกให้ผู้เรียนมองเห็นทางเลือกในการเขียนโค้ด และหลักการลำดับการคิดที่ว่านี้ ก็สามารถนำไปประยุกต์ใช้กับ IF – ELSE IF ได้เช่นกัน ดังแสดงในแบบฝึกหัดข้างล่าง

11. จงใช้ชุดโค้ดกับปัญหาคำนวณภาษีด้วยการใช้ IF – ELSE IF แต่เปลี่ยนลำดับการพิจารณาจากกลุ่มที่ (4) ย้อนกลับไปหากกลุ่มที่ (1) และพยายามลดการใช้เงื่อนไขที่ไม่จำเป็นด้วย

แบบฝึกหัดชุดที่ 6: หัดเลือกใช้ IF ที่เหมาะสม

คำชี้แจง ความเทียบเท่ากันของการใช้เงื่อนไขทำให้เรามีทางเลือกในการทำงานมากขึ้นก็จริง แต่ทางเลือกบางอย่างมันก็มาพร้อมกับความเยิ่นเย้อ ในแบบฝึกหัดนี้เราจะลองเปรียบเทียบวิธีการทำงานสองแบบจากปัญหาเดียวกัน ก่อนหน้านี้เราอาจจะคิดว่าการรวม IF – ELSE เข้าด้วยกันทำให้เราตัดทอนเงื่อนไขที่ไม่จำเป็นออกไปได้ และเราอาจจะคิดว่า เราควรจะรวมมันไว้ตลอด แต่ความจริงไม่ได้เป็นเช่นนั้น เพราะธรรมชาติของปัญหาที่เงื่อนไขการพิจารณาแยกประเภทนั้นเป็นอิสระจากกัน เราก็ควรที่จะแยก IF ออกจากกันตามธรรมชาติของปัญหา ในแบบฝึกหัดนี้เราจะเริ่ม

จากวิธีที่เหมาะสมกับการแยกออก IF ออกจากกัน และเราจะลองแก้ปัญหาคำเดิมด้วยการรวม IF เข้าด้วยกัน สำหรับปัญหา
อันหลังจะทำให้เราเห็นว่า การรวม IF ของเงื่อนไขที่เป็นอิสระจากกันจะทำให้เงื่อนไขที่ต้องใช้ทวีความซับซ้อนขึ้น
อย่างรวดเร็ว ดังนั้นเรามีความจำเป็นที่จะต้องพิจารณาความสัมพันธ์ของเงื่อนไขต่าง ๆ ให้ดีเสียก่อน

12. โปรแกรมโปรแกรมหนึ่งรับตัวเลขจำนวนเต็มหนึ่งค่ามาจากผู้ใช้ ถ้าตัวเลขนั้นหารด้วย 3 ลงตัว โปรแกรมจะพิมพ์
ข้อความว่า '3' และถ้าหารด้วย 5 ลงตัว จะพิมพ์คำว่า '5' ออกมา นั่นหมายความว่าถ้าหารด้วยทั้ง 3 และ 5
ลงตัวก็จะพิมพ์ตัวเลข 3 และ 5 ออกมา (ลำดับของการปรากฏของตัวเลขที่พิมพ์ออกมาเป็นตัวไหนก่อนก็ได้ ให้
ผลเหมือนกัน) จงเขียนโปรแกรมและชุดโค้ดของการทำงานของโปรแกรมนี้ด้วยการใช้ IF สองชุดแยกเป็นอิสระ
จากกัน [วิธีนี้เป็นวิธีที่ง่าย ถ้าทำถูกโปรแกรมจะดูกะทัดรัดสวยงาม]

13. เช่นเดียวกับข้อก่อนหน้า แต่ให้ลองเขียนโปรแกรมและชุดโค้ดของการทำงานด้วยการใช้โครงสร้าง IF และ
ELSE IF แทน คือห้ามแยก IF ออกจากกันเป็นสองส่วน [คำตอบเป็นไปได้หลายแบบ แต่มีแนวโน้มที่จะยากขึ้น
กว่าวิธีในข้อก่อนหน้าทั้งสิ้น]

แบบฝึกหัดชุดที่ 7: การใช้เงื่อนไขที่ซับซ้อนในการแก้ปัญหา

คำชี้แจง ในแบบฝึกหัดที่ผ่านมา เงื่อนไขที่เราใช้อยู่ในรูปแบบพื้นฐาน คือเราเปรียบเทียบค่าเพียงคู่เดียวก็ได้คำตอบที่ต้องการ อย่างไรก็ตาม ปัญหาจำนวนมากไม่อาจจะแก้ได้ด้วยการพิจารณาค่าเพียงคู่เดียว แต่ต้องพิจารณาค่าหลาย ๆ คู่ จึงจะได้คำตอบตามที่ต้องการ แบบฝึกหัดนี้ต้องการฝึกให้ผู้เรียนได้พิจารณาเงื่อนไขสำหรับการตัดสินใจและแยกประเภทที่ซับซ้อนขึ้น และเพื่อช่วยเหลือผู้เรียนให้มองเห็นความซับซ้อนและวิธีแก้ปัญหาไปพร้อม ๆ กัน แบบฝึกหัดจึงได้เตรียมโครงร่างของคำตอบไว้ให้บางส่วน ผู้เรียนจะต้องพิจารณาปัญหาและโครงร่างของคำตอบที่ได้ จากนั้นจึงเติมคำในชุดโค้ดที่ให้นี้ เพื่อให้ได้โค้ดที่ทำงานตามข้อกำหนด

14. จากปัญหา “จองรถตู้” ในสัปดาห์ที่แล้ว หากเรากำหนดให้ day1, day2, และ day3 คือวันที่รถตู้ A, B, และ C วางพร้อมให้ใช้งานตามลำดับ เช่น ถ้า day1 = 5, day2 = 3 และ day3 = 3 เราจะเลือกรถตู้ B. ทว่าในสถานการณ์จริง เราจะพบว่าค่าต่าง ๆ สามารถเปลี่ยนเป็นอย่างอื่นก็ได้ แต่ชื่อตัวแปรสำหรับเก็บค่าเหล่านั้นยังคงเดิม. นอกจากนี้ สิ่งที่เราต้องการก็คือวิธีการที่ตัดสินใจได้อย่างถูกต้องพร้อมทั้งรองรับค่า day1, day2, และ day3 ทุกค่าที่เป็นได้ การรู้จักตั้งเงื่อนไขการพิจารณาค่าอย่างรัดกุมจึงเป็นสิ่งที่สำคัญ และเป็นเป้าหมายของแบบฝึกหัดข้อนี้

จงเติมเงื่อนไขในชุดโค้ดต่อไปนี้เพื่อทำให้เกิดการเลือกรถตามข้อกำหนดที่ให้ไว้ก่อนหน้า (แบบฝึกหัดข้อนี้สนใจเฉพาะเงื่อนไขในการเลือกรถ เราจึงไม่ต้องคำนวณการเปลี่ยนแปลงวันที่รถจะวางให้ใช้งานในรอบถัดไป)

```
IF day1 ≤ day2 AND _____ THEN
    PRINT 'A'
ELSE IF _____ THEN
    PRINT 'B'
ELSE
    PRINT 'C'
END IF
```

คราวนี้ลองสมมติว่าชุดโค้ดเปลี่ยนเป็นแบบข้างล่าง เงื่อนไขที่เราควรเติมเข้าไปควรจะเป็นอย่างไร

```
IF _____ THEN
    PRINT 'C'
ELSE IF _____ THEN
    PRINT 'B'
ELSE
    PRINT 'A'
END IF
```

15. จากปัญหา “คัดเลือกนักเรียน” ในสัปดาห์ที่แล้ว หากกำหนดค่า K มาให้ และกำหนดเพิ่มเติมว่า $K1$ และ $K2$ คือจำนวนผู้เข้าประกวดที่กรรมการคนที่หนึ่งและคนที่สองรับไปแล้ว (สมมติว่า $0 \leq K1, K2 \leq K$) หากกำหนดให้เพศของผู้เข้าประกวดคือ G และคะแนนที่ได้จากกรรมการคนแรกคือ $S1$ ส่วนคะแนนจากกรรมการคนที่สองคือ $S2$

จงเติมเงื่อนไขของ IF ให้สมบูรณ์เพื่อหาว่าผู้เข้าประกวดจะอยู่กับกรรมการท่านใด โดยเก็บค่าไว้ในตัวแปร R โดยกำหนดให้ R มีค่าเป็น 1 เมื่อได้อยู่กับกรรมการคนแรก และมีค่าเป็น 2 เมื่อได้อยู่กับกรรมการคนที่สอง และในกรณีที่ผู้เข้าประกวดไม่ผ่านการคัดเลือก R จะมีค่าเป็น 0 (ในแบบฝึกหัดนี้ ให้สนใจเฉพาะผลการคัดเลือก ยังไม่ต้องสนใจการปรับค่าสำหรับการคำนวณแบบต่อเนื่อง)

```

IF S1 >= 9 AND G == _____ AND K1 < _____ THEN
    R = 1
ELSE IF S2 >= 9 AND _____ AND _____ THEN
    R = 2
ELSE IF S1 >= 9 AND _____ THEN
    R = 1
ELSE IF S2 >= 9 AND _____ THEN
    R = 2
ELSE
    R = 0
END IF

```

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สาม

การวางแผนการคำนวณด้วยโฟลวชาร์ตและซูโดโค้ด: การใช้โครงสร้างการทำซ้ำ

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

คำสั่ง

จงเขียนโฟลวชาร์ตที่แสดงลำดับการทำงานในแบบฝึกหัดแต่ละข้อต่อไป ในบางข้อต้องเขียนซูโดโค้ดหรือภาษาซีที่เทียบเท่ากันด้วย ทั้งนี้ขอให้ผู้เรียนเขียนโค้ดที่เทียบเท่ากับโฟลวชาร์ตที่ได้ด้วย ไม่เช่นนั้นจะถือว่าผิดในแบบฝึกหัดนี้

แบบฝึกหัดชุดที่ 1: ฝึกการวนซ้ำเพื่อนับจำนวนที่ต้องการ

คำชี้แจง แบบฝึกหัดชุดนี้มีการใช้โครงสร้างการวนซ้ำ (ลูป) โปรแกรมที่มีการวนซ้ำโดยทั่วไปจะมีงานที่ต้องทำซ้ำซึ่งจะอยู่ในลูป และมีงานที่ไม่ได้ทำซ้ำซึ่งควรอยู่นอกลูป ในแบบฝึกหัดชุดนี้เราจะเรียนรู้ความแตกต่างระหว่างงานที่ต้องทำซ้ำกับงานที่ไม่ต้องทำซ้ำ และทำการสังเกตตำแหน่งของงานที่เราใส่ไว้ในโฟลวชาร์ตหรือซูโดโค้ด

1. จงเขียนโฟลวชาร์ตของโปรแกรมที่รับจำนวนเต็มบวก K จากผู้ใช้ จากนั้นโปรแกรมจะรับจำนวนเต็มมาจากผู้ใช้อีก 100 ตัว เมื่อเสร็จสิ้นการรับข้อมูลเข้าทั้งหมด โปรแกรมจะรายงานมาจากข้อมูลเข้า 100 ตัวดังกล่าว มีกี่ตัวที่หารด้วย K ลงตัว

คำแนะนำ ในโปรแกรมควรจะมีตัวนับสองตัว ตัวหนึ่งนับไปว่าโปรแกรมรับข้อมูลเข้าก็รอบแล้ว และอีกตัวเอาไว้ระบุว่าจำนวนที่หารด้วย K ลงตัวมีเท่าใด นอกจากนี้เราควรกำหนดค่าเริ่มต้นให้กับตัวนับเหล่านี้ไว้ให้เป็น 0 ด้วย

ขอให้สังเกตด้วยว่าการรับค่า K เป็นสิ่งที่รับเพียงครั้งเดียว ดังนั้นการรับค่า K เป็นสิ่งที่ควรอยู่นอกลูป ในขณะที่การรับข้อมูลเข้าอีก 100 ตัวถัดมาเป็นสิ่งที่เราทำซ้ำ และยังเป็นการทำซ้ำในลักษณะเดิมด้วย เราจึงควรวางตำแหน่งของการอ่านข้อมูลเหล่านี้ไว้ในลูป

2. จงเขียนซูโดโค้ดหรือภาษาซีที่เทียบเท่ากับโฟลวชาร์ตในข้อที่แล้ว

=====
คำชี้แจง จากสองข้อที่แล้วเรามีการกำหนดจำนวนข้อมูลเข้าที่แน่นอนตายตัวไว้ว่าเราจะวนซ้ำ 100 รอบ และตัวแปรที่ใช้รับจำนวนที่ต้องการก็มีเพียงหนึ่งตัว แต่ในข้อต่อไป เราจะเรียนรู้การดัดแปลงโปรแกรมเพื่อให้สามารถทำงานที่ซับซ้อนกว่าเดิม

3. จงเขียนโฟลวชาร์ตของโปรแกรมที่รับจำนวนเต็มบวก N จากผู้ใช้ จากนั้นโปรแกรมจะรับจำนวนเต็มจากผู้ใช้อีก N ตัว (กล่าวคือ ค่า N ที่ผู้ใช้ใส่เข้ามาตอนแรกคือจำนวนตัวเลขที่ผู้ใช้จะใส่ตามมานั่นเอง) จากนั้นโปรแกรมจะนับว่าจากเลขจำนวน N ตัวดังกล่าวมีกี่ตัวที่หารด้วย 3 ลงตัว และมีกี่ตัวที่หารด้วย 5 ลงตัว และรายงานผลการนับออกมาเป็นผลลัพธ์ออกมาเป็นเลขสองค่า

เช่น ถ้าผู้ใช้ใส่ชุดตัวเลขมาเป็น 8 9 10 7 15 12 2 27 4 เลข 8 เป็นเลขตัวแรกจึงรับหน้าที่เป็นค่า N ดังนั้นผู้ใช้

จึงใส่เลขตามมาอีกแปดตัว และเราจะเห็นได้ว่าจากเลขทั้งแปด มีเลขที่หารด้วย 3 ลงตัวสี่ตัวคือ 9 15 12 และ 27 ส่วนเลขที่หารด้วย 5 ลงตัวมีสองตัวคือ 10 และ 15 ดังนั้นโปรแกรมจึงรายงานผลการนับมาเป็นเลข 4 และ 2

4. จงเขียนซูโดคู้ดหรือภาษาซีที่เทียบเท่ากับโพลวชาร์ตในข้อที่แล้ว

=====

คำชี้แจง จากสองข้อที่แล้วเราทราบว่าโปรแกรมจะวนซ้ำ N รอบ โดยจำนวนรอบนี้ถูกกำหนดไว้โดยชัดเจนจากผู้ใช้อย่างไรก็ตาม ในปัญหาจำนวนมาก เราจะไม่ทราบจำนวนรอบล่วงหน้าก่อนเข้าทำการวนซ้ำ แต่โปรแกรมจะหยุดทำการวนซ้ำด้วยเงื่อนไขบางอย่าง แบบฝึกหัดสองข้อต่อไปนี้จะเป็นที่ตั้งที่ทำให้เราเรียนรู้วิธีหยุดการวนซ้ำจากเงื่อนไขบางอย่าง ขอให้สังเกตด้วยว่า โปรแกรมที่อยู่ในรูปโพลวชาร์ตจะดูชัดเจนและเป็นธรรมชาติในเรื่องเกี่ยวกับการหยุดวนซ้ำ ในขณะที่โปรแกรมที่อยู่ในรูปของซูโดคู้ดจะมีโครงสร้างที่ดูซับซ้อนและเข้าใจยากกว่า

5. จงเขียนโพลวชาร์ตของโปรแกรมที่รับจำนวนเต็มจากผู้เข้ามามีค่า โดยโปรแกรมจะหยุดรับข้อมูลเมื่อผู้ใช้ใส่เลข 0 เข้ามา เมื่อหยุดรับข้อมูลแล้ว โปรแกรมจะรายงานที่ได้รับจำนวนเต็มที่ไม่ใช่ 0 จากผู้เข้ามาทั้งหมดกี่ค่า เช่น ถ้าผู้ใช้ใส่ชุดตัวเลขเข้ามาเป็น 5 7 2 0 โปรแกรมจะรายงานว่า 3 แต่ถ้าหากผู้ใช้ใส่ชุดตัวเลขมาเป็น 2 7 -9 5 -3 0 โปรแกรมจะรายงานว่า 5 และถ้าผู้ใช้ใส่เลข 0 เข้ามาเป็นค่าแรกในชุดตัวเลข โปรแกรมจะรายงานว่า 0 เป็นต้น

6. จงเขียนซูโดคู้ดหรือภาษาซีที่เทียบเท่ากับโพลวชาร์ตในข้อที่แล้ว

คำแนะนำ เราควรใช้คำสั่ง BREAK เพื่อสั่งให้หยุดการวนซ้ำ ทั้งนี้โครงสร้างของโค๊ดนี้เป็นหนึ่งในพื้นฐานที่สำคัญของการเขียนโปรแกรม หากผู้เรียนเข้าใจโครงสร้างนี้ จะเริ่มเขียนโปรแกรมได้หลากหลายขึ้นกว่าเดิมมาก

แบบฝึกหัดชุดที่ 2: ฝึกการวนซ้ำกับปัญหาที่ซับซ้อนขึ้น

คำชี้แจง แบบฝึกหัดชุดนี้รวมเทคนิคพื้นฐานต่าง ๆ เข้าด้วยกัน และแนะนำให้เห็นถึงเทคนิคใหม่อื่น ๆ พร้อมทั้งสอนให้ผู้เรียนเข้าใจวิธีการอ่านตัวอย่างข้อมูลเข้าและผลลัพธ์อันเป็นกระบวนการมาตรฐานในการอธิบายข้อกำหนดของปัญหา

7. จงเขียนโพลวชาร์ตของโปรแกรมที่รับตัวเลขจำนวนเต็มมาจากผู้ใช้จนกว่าจะเจอเลขที่ติดลบ โปรแกรมนี้ยังทำการนับด้วยว่ามีตัวเลขที่หารด้วย 3 ลงตัวเท่าใด และหารด้วย 5 ลงตัวเท่าใด และรายงานผลการนับออกมาทางเป็นค่าสองค่า ค่าแรกคือจำนวนตัวเลขที่หารด้วย 3 ลงตัว และค่าที่สองคือจำนวนตัวเลขที่หารด้วย 5 ลงตัว

[ตัวอย่างข้อมูลเข้าและผลลัพธ์อยู่ในหน้าถัดไป]

ตัวอย่างข้อมูลเข้าและผลลัพธ์

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
2	5	3	2	25	0	-12	0
10	3	7	0	-2	1		0
9		21					
6		-1					
12							
0							
30							
-9							

หมายเหตุ 0 เป็นเลขที่หารลงตัวด้วย 3 และ 5 เพราะ 0 หารด้วย 3 และ 5 แล้วเหลือเศษศูนย์

8. จงเขียนซูโดคัทหรือภาษาซีที่เทียบเท่ากับโพลวชาร์ตในข้อที่แล้ว

9. จงเขียนโพลวชาร์ตของโปรแกรมที่รับจำนวนเต็มจากผู้ใช้งานจนกว่าจะพบเลขศูนย์ ในระหว่างการรับจำนวนเต็มจากผู้ใช้งาน โปรแกรมจะทำการหาผลบวกของเลขคู่เก็บไว้ในตัวแปรชื่อ Even และทำการหาผลบวกของเลขคี่เก็บไว้ในตัวแปรชื่อ Odd หลังจากโปรแกรมได้รับเลขศูนย์ ก็จะพิมพ์ผลลัพธ์ต่อไปนี้ออกมาเป็นคำตอบตามลำดับ

(1) ค่าของ Even [ผลบวกของเลขคู่], (2) ค่าของ Odd [ผลบวกของเลขคี่], และ (3) รายงานว่า Even หรือ Odd ใครมีค่ามากกว่ากัน ถ้า Even มีค่ามากกว่าจะพิมพ์คำว่า Even แต่ถ้า Odd มีค่ามากกว่าจะพิมพ์ว่า Odd ในกรณีที่มีค่าเท่ากันจะพิมพ์ว่า Equal

หมายเหตุ ผู้เรียนต้องแยกให้ได้ว่าเมื่อจะพิมพ์ค่าของตัวแปรชื่อ Even เราจะใช้คำสั่ง Print Even แต่ถ้าต้องการพิมพ์คำว่า Even ออกมาโดยตรง (ไม่ใช่ค่าของตัวแปร) เราจะใช้คำสั่ง Print "Even" ดังนั้นตอนรายงานผลส่วนที่สามเราจะใช้คำสั่งว่า Print "Even" หรือ Print "Odd" หรือ Print "Equal" ขึ้นอยู่กับผลรวมที่ได้

ตัวอย่างข้อมูลเข้าและผลลัพธ์

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
2	12	3	2	15	0	12	0
4	20	2	-1	-11	0	-10	-5
9	Odd	7	Even	-1	Equal	-5	Even
6		-11		-3		-2	
11		0		0		0	
0							

10. จงเขียนซูโดโค้ดหรือภาษาซีที่เทียบเท่ากับโพลวชาร์ตในข้อที่แล้ว

แบบฝึกหัดชุดที่ 3: การวนซ้ำในปัญหาที่จำลองจากเหตุการณ์จริง

คำชี้แจง แบบฝึกหัดชุดนี้จะใช้เทคนิคต่าง ๆ มาแก้ปัญหาที่จำลองจากเหตุการณ์จริง เราจะเห็นได้ว่าในบางกรณี เทคนิคที่ใช้ไม่ได้ซับซ้อน แต่การวิเคราะห์สถานการณ์จากปัญหาอาจเป็นอุปสรรคในการสร้างกระบวนการคิดที่ถูกต้อง เนื่องจากเราเรียนเขียนโปรแกรมเพื่อแก้ปัญหาในการทำงานจริง เราจึงมีความจำเป็นที่จะต้องฝึกฝนการแก้ปัญหาเหล่านี้เพื่อเป็นพื้นฐานในการแก้ปัญหาที่ซับซ้อนยิ่งขึ้น ซึ่งปัญหาเหล่านั้นเป็นสิ่งที่เรามักจะพบในการประกอบอาชีพด้านคอมพิวเตอร์

11. จงเขียนโพลวชาร์ตและซูโดโค้ดที่เทียบเท่ากันของปัญหา “วันที่ตัดเสื้อแล้วเสร็จ” ซึ่งเป็นแบบฝึกหัดในสัปดาห์แรก [หมายเหตุ หากเราใช้วิธีคิดที่ดี เราจะพบว่าโครงสร้างของโปรแกรมน่าจะง่ายมาก แต่ความรู้เรื่องวิธีปัดเศษขึ้น อาจจะเป็นสิ่งที่ทำให้เรารู้สึกว่าข้อนี้เป็นข้อที่ยาก]

=====

คำชี้แจง นับจากจุดนี้ไป เราจะเริ่มพึ่งพาโพลวชาร์ตน้อยลงเรื่อย ๆ และจะใช้ซูโดโค้ดเพื่อวางแผนการคิด เพื่อเตรียมตัวสู่กระบวนการเขียนโปรแกรมด้วยภาษาคอมพิวเตอร์อันเป็นเป้าหมายหลักของวิชานี้ ดังนั้นแบบฝึกหัดสองข้อต่อไปนี้จะไม่ระบุให้เขียนโพลวชาร์ต และผู้เรียนสามารถตัดสินใจด้วยตนเองได้ว่าจะเขียนโพลวชาร์ตก่อนหรือไม่ หรือจะสร้างกระบวนการคิดโดยไม่ต้องใช้โพลวชาร์ต

12. จงเขียนซูโดโค้ดของปัญหา “จองรถตู้” จากแบบฝึกหัดสัปดาห์แรก ทั้งนี้ผู้เรียนอาจใช้ประโยชน์จากซูโดโค้ดที่คิดไว้ในสัปดาห์ที่สองเพื่อนำมาดัดแปลงให้เป็นโค้ดที่สมบูรณ์สำหรับปัญหานี้ได้

13. จงเขียนซูโดโค้ดของปัญหา “คัดเลือกนักเรียน” จากแบบฝึกหัดสัปดาห์แรก ทั้งนี้ผู้เรียนอาจใช้ประโยชน์จากซูโดโค้ดที่คิดไว้ในสัปดาห์ที่สองเพื่อนำมาดัดแปลงให้เป็นโค้ดที่สมบูรณ์สำหรับปัญหานี้ได้

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สี่

เตรียมตัวสอบย่อยด้วยตัวอย่างข้อสอบย่อยปีการศึกษา 2556 ภาคการศึกษาต้น

คำสั่ง (1) ให้เขียนคำตอบลงในกระดาษคำถามตรงเนื้อที่ที่เว้นไว้ให้ แต่หากที่ไม่พอให้ใช้พื้นที่ด้านหลังได้ โดยให้ระบุให้ชัดเจนว่าคำตอบอยู่ทางด้านหลัง (2) ตอบด้วยดินสอ 2B ขึ้นไปได้ (3) ใช้พื้นที่ด้านหลังทดเลขได้ (4) ห้ามฉีกข้อสอบออกจากกัน (5) อนุญาตให้ออกจากห้องสอบได้หากเวลาผ่านไปเกิน 30 นาทีนับจากเริ่มสอบ

คะแนน ข้อ 1 _____ ข้อ 2 _____

ข้อ 3 a _____ b _____ c _____ d _____ e _____

1. จงแสดงวิธีทำเพื่อหาค่า x และ y จากสมการสองตัวแปรทางด้านล่าง

$$3x = 2y + 1 + x$$

$$5y + 3x = 2x + 2$$

(ตอบเป็นเลขในรูปเศษหรือเลขทศนิยมก็ได้) [6 คะแนน]

2. รถยนต์คันหนึ่งวิ่งไปด้วยความเร็ว 40 กิโลเมตรต่อชั่วโมงเป็นเวลา 15 นาที จากนั้นจึงเปลี่ยนไปวิ่งที่ความเร็ว 60 กิโลเมตรต่อชั่วโมงเป็นเวลา 20 นาที ต่อมาก็เปลี่ยนความเร็วอีกครั้งเป็น 90 กิโลเมตรต่อชั่วโมงเป็นเวลา 45 นาที จงหาว่า รถยนต์วิ่งไปเป็นระยะทางรวมเท่าใด (แสดงวิธีคิดสั้น ๆ ด้วยว่าคำนวณคำตอบแต่ละส่วนมาได้อย่างไร) [4 คะแนน]

3. มหาวิทยาลัยแห่งหนึ่งตั้งระดับผลการเรียน (เกรด) ไว้สามระดับคือ 0, 1, และ 2 คะแนนในวิชาฟิสิกส์ของมหาวิทยาลัยแห่งนี้แบ่งออกเป็นสองส่วนคือคะแนนสอบย่อย (x), คะแนนสอบกลางภาค (y) และ คะแนนสอบปลายภาค (z) ถ้าหากนักศึกษาได้ผลรวมคะแนนสอบทั้งสามส่วนนี้ต่ำกว่า 40 คะแนนจะได้เกรดเป็น 0 ถ้าถึง 40 คะแนนแต่ไม่ถึง 80 คะแนนได้เกรดเป็น 1 และหากได้ 80 คะแนนขึ้นไป ได้เกรดเป็น 2

ตัวอย่าง

คะแนนสอบย่อย	คะแนนสอบกลางภาค	คะแนนสอบปลายภาค	เกรด	คะแนนสอบย่อย	คะแนนสอบกลางภาค	คะแนนสอบปลายภาค	เกรด
20	40	40	2	5	10	20	0
10	20	10	1	5	15	30	1

- a) จงเขียนฟังก์ชันคณิตศาสตร์ f ที่ขึ้นกับค่า x, y และ z สำหรับคำนวณเกรดที่จะได้ [10 คะแนน] **หมายเหตุ** ถ้าไม่สามารถเขียนฟังก์ชันได้ แต่อธิบายการคำนวณได้ถูกต้อง จะได้ 5 คะแนน

- b) เนื่องจากนักศึกษาต่างคาดหวังที่จะได้ระดับผลการเรียนที่ดี ทางอาจารย์จึงได้ทำการคำนวณว่าแต่ละคนจะต้องทำคะแนนเพิ่มอีกเท่าใดจึงจะได้เกรดที่ดีขึ้นอีกหนึ่งระดับ โดยหากนักศึกษาที่ได้เกรดเป็น 0 อาจารย์จะแจ้งคะแนนที่นักศึกษาคนดังกล่าวต้องทำเพิ่มเพื่อให้ได้เกรดเป็น 1 และถ้านักศึกษาได้เกรดเป็น 1 อาจารย์จะแจ้งคะแนนที่ต้องทำเพิ่มเพื่อให้ได้เกรดเป็น 2 แต่ถ้านักศึกษาได้เกรดเป็น 2 อาจารย์จะแจ้งคะแนนที่ต้องทำเพิ่มว่าเท่ากับ 0 คะแนน จงเขียนฟังก์ชัน g ที่ขึ้นกับค่า x, y , และ z ที่ทำการคำนวณว่าจากคะแนนที่ได้ นักศึกษาจะต้องทำคะแนนเพิ่มอีกเท่าใดจึงจะได้เกรดที่ดีขึ้นอีกหนึ่งระดับ (แนะนำให้ใช้ผลจากฟังก์ชัน f เป็นตัวช่วยในการคำนวณ) [10 คะแนน]

c) กำหนดข้อมูลเข้าคือคะแนนสอบสามค่าที่ถูกแทนด้วย x , y และ z ตามลำดับ จงเขียนโปรแกรมแสดงการคำนวณเกรดของนักศึกษาที่ได้ และคะแนนที่นักศึกษาต้องทำเพิ่มเพื่อให้ได้เกรดที่ดีขึ้นอีกหนึ่งระดับ [15 คะแนน] (ถ้าที่เขียนไม่พอ เขียนคำตอบไว้ด้านหลังของกระดาษคำตอบได้)

d) จงเขียนชุดโค้ดหรือโค้ดภาษาซีอย่างใดอย่างหนึ่งที่เทียบเท่ากับโปรแกรมที่ได้ในข้อ c) [15 คะแนน] [หากโค้ดทำงานถูกต้อง แต่โค้ดไม่เทียบเท่ากับโปรแกรมจะได้ 12 คะแนน] (เขียนตอบด้านหลังกระดาษได้)

- e) อาจารย์มีนักเรียนอยู่ N คน เขาต้องการโปรแกรมคำนวณเกรดเฉลี่ยของนักเรียนทั้งหมด โดยข้อมูลเข้าตัวแรกของโปรแกรมคือ N ต่อมาข้อมูลเข้าจะมาทั้งหมดอีก N ชุด ชุดละสามค่า คือคะแนนสอบย่อย คะแนนสอบกลางภาค และ คะแนนสอบปลายภาค หนึ่งชุดต่อนักศึกษาหนึ่งคน จงเขียนโฟลวชาร์ตของโปรแกรมคำนวณเกรดเฉลี่ย เมื่อกำหนดวิธีคิดเกรดของแต่ละคนตามที่ระบุไว้ในข้อย่อย a) [40 คะแนน]

หมายเหตุ อย่าเขียนภาษาซีหรือซูโดโค้ดมาไม่เช่นนั้นจะไม่ได้คะแนน ข้อนี้ต้องการโฟลวชาร์ตเท่านั้น ให้เขียนคำตอบไว้ที่ด้านหลังของกระดาษแผ่นนี้ อย่ารับข้อมูลเข้าหรือแสดงผลในสิ่งที่ไม่ได้บอกให้รับหรือแสดง ไม่งั้นจะโดนหักคะแนน

คำแนะนำ เนื่องจากโฟลวชาร์ตค่อนข้างยาว แนะนำให้เขียนคำตอบไว้ที่ด้านหลังของกระดาษแผ่นนี้จะดีกว่า

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
3 20 40 40 20 20 20 10 10 10	1	5 10 20 30 5 20 20 15 25 25 15 30 30 15 35 35	1.2

เฉลยข้อสอบย่อยครั้งที่หนึ่ง วิชา 517111 Computer Programming I

ปีการศึกษา 2556 ภาคการศึกษาต้น

1. $x = 3 / 4; y = 1 / 4$

2. $(40 \times \frac{15}{60}) + (60 \times \frac{20}{60}) + (90 \times \frac{45}{60}) = 10 + 20 + 67.5 = 97.5$ กิโลเมตร

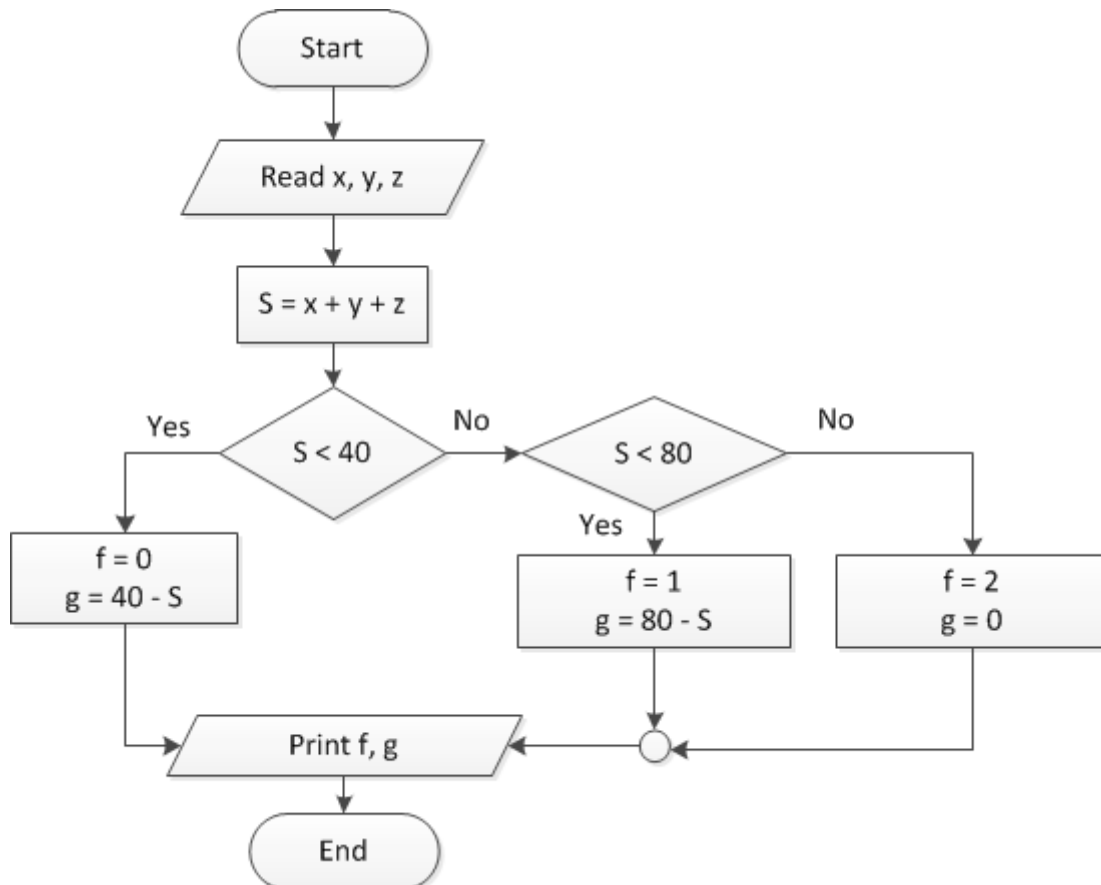
3. a) $f(x, y, z) = \begin{cases} 0; & x+y+z < 40 \\ 1; & 40 \leq x+y+z < 80 \\ 2; & x+y+z \geq 80 \end{cases}$ (จะใช้คำว่า *otherwise* แทนก็ได้)

b) $g(x, y, z) = \begin{cases} 40 - (x+y+z); & x+y+z < 40 \\ 80 - (x+y+z); & 40 \leq x+y+z < 80 \\ 0; & x+y+z \geq 80 \end{cases}$ (จะใช้คำว่า *otherwise* แทนก็ได้)

หรือถ้าเขียนแบบใช้ f เป็นตัวช่วยก็จะได้เป็น

$g(x, y, z) = \begin{cases} 40 - (x+y+z); & f(x, y, z) = 0 \\ 80 - (x+y+z); & f(x, y, z) = 1 \\ 0; & f(x, y, z) = 2 \end{cases}$ (จะใช้คำว่า *otherwise* แทนก็ได้)

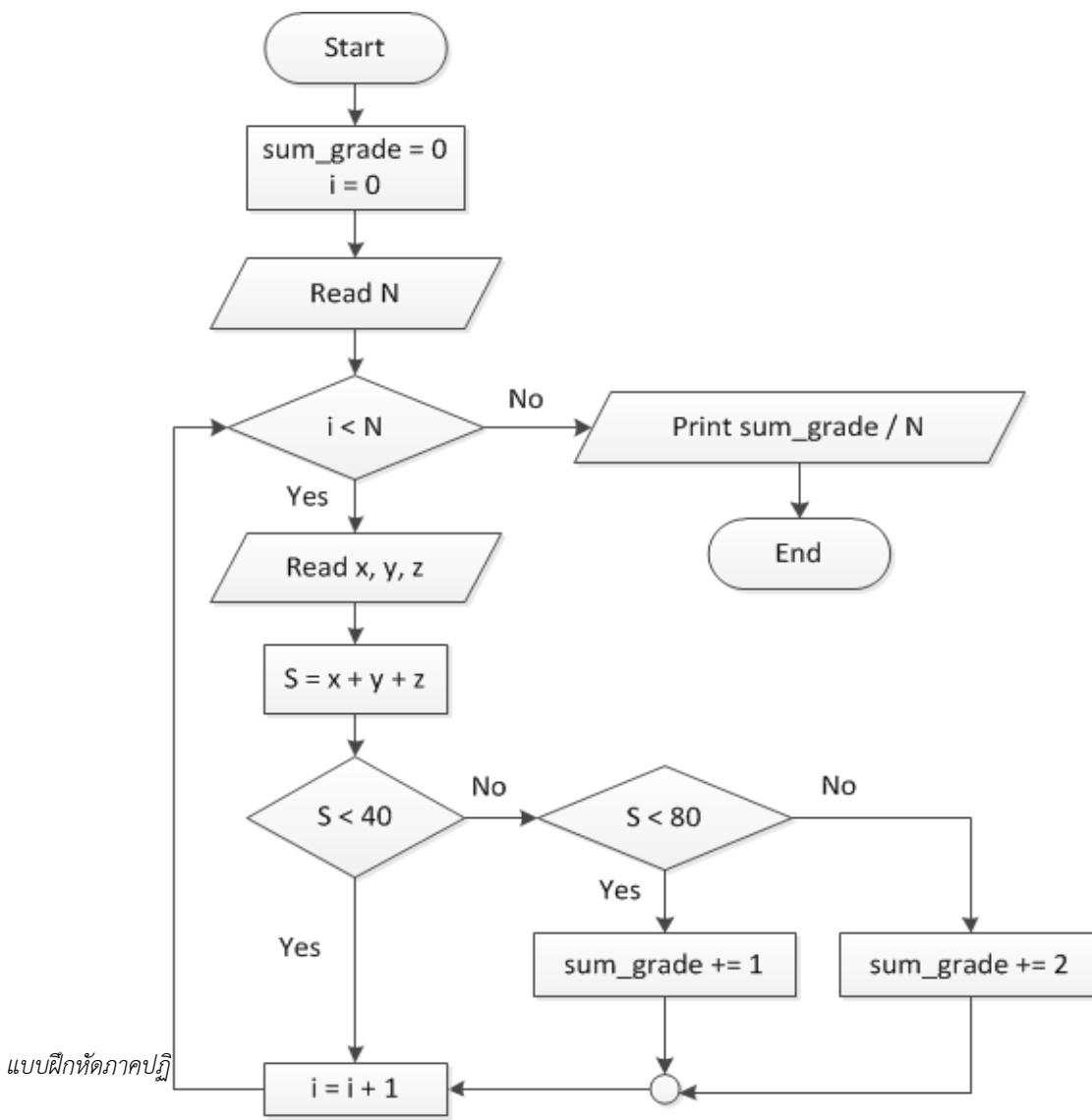
c)



d)

```
START
READ x, y, z
S = x + y + z
IF S < 40 THEN
  f = 0
  g = 40 - S
ELSE IF S < 80 THEN
  f = 1
  g = 80 - S
ELSE
  f = 2
  g = 0
END IF
PRINT f, g
END
```

e)



แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่ห้า

พื้นฐานการเขียนภาษาซี เงื่อนไขการทำงาน

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

คำสั่ง

1. ให้เขียนโปรแกรมภาษาซีสำหรับปัญหาที่ให้ไป
2. ระเบียบการส่งงานเขียนโปรแกรมก็คือนักศึกษาจะส่งโค้ดเข้าไปโปรแกรมตรวจงาน (เปิดใช้สัปดาห์หน้า) ในสัปดาห์นี้เราจะฝึกการทดสอบโปรแกรมด้วยตัวเอง ส่วนในสัปดาห์ถัดไปเราจะทดสอบโปรแกรมด้วยตนเอง เมื่อคิดว่าถูกต้องแล้ว เราจึงลองส่งไปตรวจด้วยระบบตรวจอัตโนมัติที่ผู้สอนเตรียมไว้

ปัญหาชุดที่ 1: พื้นฐานการรับข้อมูลเข้า การคำนวณเลขคณิต และการแสดงผล

จุดประสงค์การเรียนรู้ 1: เพื่อให้ผู้เรียนรู้จักพื้นฐานการใช้งานคำสั่ง printf ในการพิมพ์ข้อความที่ต้องการ แบบฝึกหัดนี้ไม่จำเป็นต้องรับข้อมูลเข้าหรือประกาศตัวแปรแต่อย่างใด

ปัญหา 1 เกี่ยวกับตัวคุณ [about_you]

จงเขียนโปรแกรมที่พิมพ์รหัสประจำตัวนักศึกษาในบรรทัดแรก และพิมพ์ชื่อ-นามสกุลนักศึกษาในบรรทัดที่สองเป็นภาษาอังกฤษ เช่น

07551234

Pinyo Taeprasartsit

หมายเหตุ ข้อนี้นักศึกษาไม่จำเป็นต้องส่งคำตอบไปตรวจที่ระบบตรวจคำตอบ

=====

จุดประสงค์การเรียนรู้ 2: เพื่อให้ผู้เรียนฝึกรับข้อมูลเข้าและแสดงผลลัพธ์จากตัวแปรที่เก็บข้อมูลเข้าที่ต้องการจากผู้เรียนจะได้เรียนรู้การใช้คำสั่ง scanf และ printf ไปด้วยกัน

ปัญหา 2 พิมพ์ฉัน [print_me]

จงเขียนโปรแกรมที่รับจำนวนเต็มสองจำนวน คือ x และ y จากนั้นโปรแกรมจะพิมพ์เลขทั้งสองออกมาตามลำดับในบรรทัดเดียวกันและคั่นด้วยช่องว่างหนึ่งช่อง เช่น ถ้าผู้ใช้ใส่ค่า x และ y มาเป็น 3 และ 5 ตามลำดับ สิ่งที่โปรแกรมจะพิมพ์ออกมาเป็น 3 5 [มีตัวอย่างข้อมูลเข้าและผลลัพธ์เพิ่มเติมอยู่ในหน้าถัดไป]

ข้อควรจำ การใช้คำสั่ง scanf เรากำหนดรูปแบบข้อมูลเข้าภายในเครื่องหมายอัฒภาคคู่ “ ” และเราสามารถใส่เฉพาะสิ่งที่เกี่ยวข้องกับตัวแปร (เช่น %d) และช่องว่าง (space หรือ tab) ได้เท่านั้น เช่น scanf(“%d %d”, &x, &y);

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
5 3	5 3	5 3	5 3	-7 8	-7 8

เรื่อนำรู้

ข้อมูลเข้าแบบจำนวนเต็มและเลขทศนิยมแต่ละตัวจะสิ้นสุดเมื่อโปรแกรมพบช่องว่าง (space หรือ tab) หรือการขึ้นบรรทัดใหม่ (space) และข้อมูลตัวเลขถัดไปจะถูกคั่นด้วยช่องว่างและการขึ้นบรรทัดใหม่ก็ตัวก็ได้ คอมพิวเตอร์จะข้ามข้อมูลเหล่านี้ให้เราโดยอัตโนมัติ ดังนั้นการใส่ข้อมูลเข้าสองตัวจึงสามารถคั่นด้วยว่าง ขึ้นบรรทัดใหม่ หรือ การเลื่อนกันหน้าก็ได้ และจะผสมกันก็ได้ นอกจากนี้การใส่ตัวอักขระพวกนี้เข้าไปหลายตัวก็ไม่ทำให้เกิดการเปลี่ยนแปลงเช่นกัน

สิ่งที่ควรระวัง

อย่าใส่ของอย่างอื่นเข้าไปโดยเฉพาะ `\n` เช่น แบบนี้ผิด `scanf("%d %d\n", &x, &y);` ซึ่งการใส่ `\n` ปนเข้าไปใน `scanf` เป็นหนึ่งในความผิดพลาดที่พบได้บ่อยที่สุดสำหรับผู้เริ่มต้น เนื่องจากสับสนกับสิ่งที่ทำใน `printf`

ปัญหา 3 พิมพ์ฉันทสอง [print_me2]

จงเขียนโปรแกรมที่รับจำนวนเต็มสองจำนวน คือ x และ y จากนั้นโปรแกรมจะพิมพ์เลขทั้งสองออกมาแบบสลับลำดับในบรรทัดแรกและคั่นด้วยช่องว่างหนึ่งช่อง เช่น ถ้าผู้ใช้ใส่ค่า x และ y มาเป็น 3 และ 5 ตามลำดับ สิ่งที่โปรแกรมจะพิมพ์ออกมาเป็น 5 3 ส่วนบรรทัดที่สามให้พิมพ์ค่า x ออกมา และบรรทัดที่สี่พิมพ์ค่า y ออกมา

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
5 3	3 5	5 3	3 5 5 3	-7 8	8 -7 -7 8

ข้อสังเกต

เราใส่ข้อมูลเข้าไปเพียงสองครั้ง แต่การแสดงผลจะนำมาแสดงก็ครั้งก็ได้ และแสดงในลำดับที่ต่างกับข้อมูลเข้าก็ได้ และสามารถจัดรูปแบบให้แตกต่างกันก็ได้เช่นกัน

=====

จุดประสงค์การเรียนรู้ 3: การจัดการกับตัวเลขเป็นสิ่งที่จำเป็นมากในการเขียนโปรแกรมเนื่องจากหนึ่งในหน้าที่หลักของโปรแกรมจำนวนมากก็คือการคำนวณสิ่งที่มนุษย์ทำได้ช้าหรือทำผิดพลาดได้ง่าย ในแบบฝึกหัดชุดนี้เราจะหัดใช้ตัวดำเนินการคณิตศาสตร์พื้นฐาน

ปัญหา 4 บวกเลขจำนวนเต็ม [add_number]

จงเขียนโปรแกรมที่รับจำนวนเต็มมาสามค่าคือ x , y , และ z จากนั้นโปรแกรมจะพิมพ์ผลบวกของเลขทั้งสามค่าออกมาทางจอภาพ

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
5 3 -2	6	1 2 3	6	0 -2 1	-1

ปัญหา 5 เลขหลักหน่วย [last_digit]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็มบวก x จากผู้ใช้ และโปรแกรมจะพิมพ์เลขหลักหน่วยออกมาเป็นผลลัพธ์

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
789	9	987	7	897	7

ปัญหา 6 เลขหลักสิบ [second_last_digit]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็มบวก x จากผู้ใช้ และโปรแกรมจะพิมพ์เลขหลักสิบออกมาเป็นผลลัพธ์ ในกรณีที่ x เป็นเลขมีค่าน้อยกว่า 10 โปรแกรมจะพิมพ์เลขศูนย์ออกมาเป็นผลลัพธ์

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
789	8	207	0	7	0

คำแนะนำ ถ้าเราหาร x ด้วย 10 ก่อน ผลลัพธ์ที่ได้จะเป็นค่า x ที่หลักหน่วยหายไป เช่น จากเลข $x = 789$ ถ้าเราหาร x ด้วย 10 ผลลัพธ์ที่ได้ก็คือ 78 ทำให้เราใช้วิธีการเดิมจากข้อที่แล้วมาช่วยในการหาเลขหลักสิบได้

เรื่องชวนคิด ถ้าหากเราใช้วิธีที่เสนอไปตรงคำแนะนำ หาก x มีค่าน้อยกว่า 10 ผลลัพธ์จะออกมาเป็นศูนย์ตามข้อกำหนดของปัญหาหรือไม่

ปัญหาชุดที่ 2: รู้จักกับจำนวนและการเปรียบเทียบค่าในคำสั่งเงื่อนไข

จุดประสงค์การเรียนรู้ 4: จากแบบฝึกหัดชุดที่แล้ว เราได้รู้จักกับการรับค่าและการดำเนินการเลขคณิตขั้นพื้นฐาน อย่างไรก็ตาม เรายังไม่เคยทำการจำแนกประเภทหรือลักษณะของข้อมูลที่พบ ซึ่งการจำแนกข้อมูลเป็นสิ่งที่สำคัญ โดยเฉพาะการกรองข้อมูลเข้าที่ผิดพลาด เช่น หากเราต้องคำนวณจำนวนนักศึกษาทุกชั้นปีรวมกัน โดยโปรแกรมคำนวณรับข้อมูลนักศึกษาแต่ละชั้นปีเป็นข้อมูลเข้า จำนวนนักศึกษาที่รับมาจากผู้ใช้ไม่ควรจะเป็นเลขติดลบ ถ้าติดลบแสดงว่าผู้ใช้กรอกข้อมูลเข้าผิดพลาด

อย่างไรก็ตาม หากเราไม่ทราบวิธีจำแนกลักษณะของข้อมูลเราจะไม่สามารถตรวจหาความผิดพลาดของข้อมูลได้อย่างที่ควรเป็น แบบฝึกหัดนี้จะฝึกให้ผู้เรียนได้รู้จักพื้นฐานการใช้โครงสร้างเงื่อนไข (IF, IF-ELSE, IF-ELSE IF) และการตั้งเงื่อนไขเพื่อจำแนกลักษณะของข้อมูล ซึ่งพื้นฐานเหล่านี้เป็นสิ่งที่จำเป็นในการประยุกต์ใช้แก้ปัญหาการคำนวณที่พบบ่อยในสถานการณ์จริง

ปัญหา 7 เลขบวก [positive]

จงเขียนโปรแกรมที่พิมพ์คำว่า positive เมื่อผู้ใช้ใส่ค่าตัวเลขจำนวนเต็มบวกมาเป็นข้อมูลเข้า และโปรแกรมนี้จะไม่พิมพ์สิ่งใดออกมาเลย ถ้าค่าตัวเลขไม่เป็นบวก [หมายเหตุ ปัญหาข้อนี้ฝึกให้นักศึกษารู้จักการใช้คำสั่งเงื่อนไข if โดยไม่ต้องใช้ else]

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
789	positive	1	positive	0	

คำชี้แจง ในตัวอย่างที่ 3 นั้น $x = 0$ ซึ่งไม่เป็นบวก โปรแกรมจึงจบการทำงานโดยไม่พิมพ์ผลลัพธ์ใดออกทางจอภาพ

ปัญหา 8 เลขคู่หรือคี่ [even_or_odd]

จงเขียนโปรแกรมที่รับจำนวนเต็ม x จากผู้ใช้ หาก x เป็นคี่ โปรแกรมจะพิมพ์คำว่า odd แต่ถ้า x เป็นคู่ โปรแกรมจะพิมพ์คำว่า even [หมายเหตุ ให้ใช้คำสั่งเงื่อนไข if-else ในการแก้ปัญหาข้อนี้]

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
9	odd	-5	odd	0	even

ปัญหา 9 บวก ลบ หรือว่า ศูนย์ [number_sign]

จงเขียนโปรแกรมด้วยการใช้โครงสร้าง if-else if ซึ่งโปรแกรมจะทำการรับค่าจำนวนเต็ม x เป็นข้อมูล หาก x เป็นบวก โปรแกรมจะพิมพ์คำว่า positive ถ้าเป็นลบจะพิมพ์คำว่า negative แต่ถ้าเป็นศูนย์ จะพิมพ์คำว่า zero

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
9	positive	-5	negative	0	zero

ปัญหา 10 อย่าหารเลขด้วยศูนย์ [no_divide_by_zero]

การหารด้วยศูนย์เป็นสิ่งที่ไม่มีนิยามในคณิตศาสตร์ และเป็นการกระทำต้องห้ามในภาษาซีและภาษาคอมพิวเตอร์อื่น ๆ อีกจำนวนมาก ดังนั้นเราต้องเลี่ยงการหารด้วยศูนย์ด้วยการตรวจสอบตัวหารก่อน เช่นในแบบฝึกหัดข้อนี้ ถ้าหากตรวจพบว่าตัวหารเป็นศูนย์ เราจะไม่ทำการหาร แต่ถ้าไม่เป็นศูนย์ เราจะทำการหารตามปกติ

จงเขียนโปรแกรมที่รับเลขทศนิยมแบบ double precision มาสามค่าคือ x , y , และ z จากนั้นถ้า z ไม่เป็นศูนย์ โปรแกรมจะทำการหาค่าของ x หาร y แล้วนำผลที่ได้หารด้วย z เสร็จแล้วให้พิมพ์ผลลัพธ์ออกมาเป็นเลขทศนิยม 6 ตำแหน่ง แต่หาก z มีค่าเท่ากับศูนย์ โปรแกรมจะไม่ดำเนินการบวกหรือหารเลข แต่จะพิมพ์ว่า

cannot divide by zero ออกมาเป็นผลลัพธ์ทันที

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
0.5 1.0 3	0.500000	0.5 1.0 0	cannot divide by zero
1 -1 5	0.000000	5 3 -2.0	-4.000000

ปัญหา 11 หาเศษและพิมพ์ผลลัพธ์ [modulo_and_printf]

[ข้อนี้เป็นคำถามต่อเนื่องมาจากโจทย์สัปดาห์ที่ 2 ข้อ 12]

จงเขียนโปรแกรมซึ่งรับตัวเลขจำนวนเต็มหนึ่งค่ามาจากผู้ใช้ ถ้าตัวเลขนั้นหารด้วย 3 ลงตัว โปรแกรมจะพิมพ์ข้อความว่า '3' และถ้าหารด้วย 5 ลงตัว จะพิมพ์คำว่า '5' ออกมา นั่นหมายความว่าถ้าหารด้วยทั้ง 3 และ 5 ลงตัวก็จะพิมพ์ตัวเลข 3 และ 5 ออกมา ในกรณีนี้ให้แยกตัวเลขทั้งสองด้วยช่องว่างหนึ่งช่อง และให้พิมพ์เลข 3 ก่อนเลข 5

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
12	3	15	3 5
25	5	7	

หมายเหตุ ในตัวอย่างสุดท้ายโปรแกรมไม่มีผลลัพธ์ใด ๆ ที่ถูกพิมพ์ออกมาทางจอภาพ

ปัญหา 12 เลือกตัวดำเนินการ [operator_selection]

จงเขียนโปรแกรมที่รับจำนวนเต็ม x , y , และ z จากผู้ใช้ ซึ่งถ้า $z = 1$ โปรแกรมจะหาค่า $x + y$ และพิมพ์ผลลัพธ์ออกมาทางจอภาพ แต่ถ้า $z = 2$ โปรแกรมจะพิมพ์ค่าของ $x - y$ ถ้า $z = 3$ โปรแกรมจะพิมพ์ค่าของ $x * y$ และถ้า $z = 4$ โปรแกรมจะพิมพ์ค่าของ $x \div y$ แต่ถ้า $z = 5$ โปรแกรมจะพิมพ์ค่าเศษจากการหาร x ด้วย y อย่างไรก็ตามการหารด้วย 0 เป็นสิ่งที่ไม่นิยาม ดังนั้น ถ้า $z = 4$ หรือ 5 และ $y = 0$ โปรแกรมจะพิมพ์คำว่า cannot divide by zero ออกมาแทน โดยไม่พยายามทำการหารหรือหาเศษจากการหารด้วย 0

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
12 3 1	15	12 3 2	9
12 3 3	36	12 3 4	4
12 3 5	0	12 0 4	cannot divide by zero
12 0 5	cannot divide by zero		

ปัญหา 13 ค่าสัมบูรณ์ [absolute_value]

จงเขียนโปรแกรมที่รับจำนวนเต็ม x จากผู้ใช้ และพิมพ์ค่าสัมบูรณ์ของ x มาเป็นผลลัพธ์ กล่าวคือ ถ้า x เป็นค่าลบก็จะกลับค่า x ให้เป็นบวกแล้วพิมพ์ค่าดังกล่าวออกมา แต่ถ้า x เป็นบวกหรือศูนย์ก็จะพิมพ์ค่า x ออกมาเป็นผลลัพธ์

โดยตรงเลย

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
12	12	0	0
-25	25	-12	12

ปัญหาชุดที่ 3: การประยุกต์ใช้การดำเนินการคณิตศาสตร์ในการแก้ปัญหา

จุดประสงค์การเรียนรู้ 5: เจาะลึกสำหรับการจำแนกประเภทหรือหาคำตอบในปัญหาที่เกี่ยวข้องกับสถานการณ์จริง มักจะค่อนข้างซับซ้อน ผู้พัฒนาโปรแกรมจำเป็นต้องมีทั้งความเข้าใจเกี่ยวกับตัวปัญหาและวิธีการเขียนโปรแกรม เป็นอย่างดีจึงจะสามารถแก้ปัญหาคำนวณเหล่านี้ด้วยโปรแกรมคอมพิวเตอร์ได้

ในแบบฝึกหัดชุดที่แล้ว เราได้ฝึกพื้นฐานการใช้คำสั่งเงื่อนไขและทดลองใช้กับตัวอย่างที่เกี่ยวข้องกับระบบ จำนวนโดยตรง ในขณะที่ทักษะที่เรียนรู้ในแบบฝึกหัดชุดที่แล้วเป็นพื้นฐานที่สำคัญ เงื่อนไขที่ใช้และสถานการณ์จัดว่า มีความซับซ้อนน้อย คือเราสามารถที่จะแก้ปัญหาในแบบฝึกหัดนั้นโดยไม่ได้ใช้ตัวดำเนินการทางตรรกะที่เชื่อมค่าตรรกะ สองค่าเข้าด้วยกัน (คือการใช้ `&&` และ `||`) ในขณะที่ปัญหาในแบบฝึกหัดชุดนี้จะมีความซับซ้อนขึ้นมาก การไม่ได้ใช้ตัวดำเนินการทางตรรกะดังกล่าว มีแนวโน้มที่จะทำให้โปรแกรมเข้าใจยากและเินเยือกกว่าที่ควรจะเป็นมาก กล่าวคือ ในแบบฝึกหัดชุดนี้จะมีหลายปัญหาที่เราควรพิจารณาการใช้ตัวดำเนินการทางตรรกะที่เหมาะสมเพื่อให้ได้โปรแกรมที่กะทัดรัด เข้าใจง่าย และมีโอกาสเกิดความผิดพลาดในการเขียนโปรแกรมลดลง

ปัญหา 14 ตัดเกรดสามระดับ [grading_fpe]

จงเขียนโปรแกรมตัดเกรด โดยโปรแกรมจะรับค่าคะแนนมาทาง standard input (keyboard) จากนั้นจะตัดเกรด โดยใช้เกณฑ์ว่า ถ้าได้ไม่ถึง 40 คะแนนได้เกรดเป็น Fail, ได้ถึง 40 แต่ไม่ถึง 80 คะแนนได้เกรดเป็น Pass, และถ้าได้ถึง 80 คะแนน จะได้เกรดเป็น Excellent

หมายเหตุ ในการพิมพ์ผลลัพธ์ออกมา โปรแกรมจะต้องพิมพ์ตัวอักษรภาษาอังกฤษเป็นตัวเล็กหรือตัวใหญ่ตามที่ระบุไว้ในตัวอย่าง ไม่เช่นนั้นจะถือว่าเป็นคำตอบที่ผิด

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์1
39	Fail	40	Pass	92	Excellent

ปัญหา 15 เรียงเบอร์สามตัว [3num_sort]

จงเขียนโปรแกรมที่รับจำนวนเต็มจากผู้เข้ามาทั้งหมด 3 ตัว คือ x, y, และ z จากนั้นโปรแกรมจะพิมพ์จำนวนทั้งสามออกมาโดยเรียงจากค่าน้อยไปหามาก

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
-8 5 3	-8 3 5	4 5 6	4 5 6
9 8 7	7 8 9	9 7 8	7 8 9
8 9 7	7 8 9	7 7 6	6 7 7

ปัญหา 16 จัดถุงยังชีพ [survival_bag]

ถุงยังชีพถุงหนึ่งประกอบไปด้วยน้ำ 3 ขวด, ขนมปัง 4 ก้อน, และ ไข่ต้ม 2 ฟอง หากกองบรรเทาทุกข์ได้รับน้ำ ขนมปัง และ ไข่ต้ม จากผู้บริจาคมาเป็นปริมาณ x ขวด, y ก้อน, และ z ฟอง ตามลำดับ กองบรรเทาทุกข์จะจัดถุงยังชีพตามข้อกำหนดข้างต้นได้สูงสุดกี่ถุง และจะเหลือของบริจาคแต่ละอย่างเป็นปริมาณเท่าใด

จงเขียนโปรแกรมที่รับปริมาณของบริจาคเป็นเลขจำนวนเต็ม x , y และ z ตามลำดับ จากนั้นโปรแกรมจะพิมพ์เลขออกมาสี่ตัว โดยเลขตัวแรกคือจำนวนถุงยังชีพที่มากที่สุดที่จัดได้ และตัวเลขสามตัวถัดมาคือปริมาณน้ำ ขนมปัง และ ไข่ต้มที่เหลือจากการผลิตตามลำดับ ให้ตัวเลขแต่ละตัวคั่นด้วยช่องว่าง 1 ช่อง หรือจะขึ้นบรรทัดใหม่ก็ได้

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
10 20 30	3 1 8 24
50 20 30	5 35 0 20
50 70 30	15 5 10 0
30 40 50	10 0 0 30
50 43 21	10 20 3 1

ปัญหา 17 จองรถตู้ [reserve_van]

จากปัญหาจองรถตู้ที่ได้กล่าวถึงในสัปดาห์แรก หากกำหนดให้ผู้ใช้ใส่เลขจำนวนเต็มบวกมาสามค่าคือ $day1$, $day2$, และ $day3$ ซึ่งแทนวันที่รถตู้ A, B, และ C จะว่างให้ใช้งาน ตามลำดับ และสิ่งที่ผู้ใช้ต้องการทราบก็คือรถตู้ที่จะถูกเลือกใช้งาน

จงเขียนโปรแกรมที่รับค่า $day1$, $day2$, และ $day3$ เป็นข้อมูลเข้า และพิมพ์ตัวอักษร A เมื่อรถตู้ A จะถูกเลือก, พิมพ์ตัวอักษร B เมื่อรถตู้ B จะถูกเลือก และพิมพ์ตัวอักษร C เมื่อรถตู้ C จะถูกเลือก

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
1 1 1	A

2 1 1	B
3 2 1	C
3 4 3	A
4 3 4	B
4 4 3	C

ปัญหา 18 คัดเลือกนักร้อง [the_voice_su]

จากปัญหาคัดเลือกนักร้องที่ได้กล่าวถึงในสัปดาห์แรก หากกำหนดให้ค่า K เป็นข้อมูลเข้าซึ่งระบุจำนวนผู้เข้าประกวดที่กรรมการแต่ละท่านจะรับไว้ได้ และกำหนดค่า K_1 และ K_2 เป็นจำนวนผู้เข้าประกวดที่กรรมการคนที่ 1 และ 2 รับไว้แล้วในปัจจุบัน กำหนดเพิ่มเติมว่า S_1 และ S_2 เป็นคะแนนที่กรรมการคนแรกและคนที่สองให้กับผู้เข้าประกวดที่โปรแกรมกำลังพิจารณา และค่า G เป็นเพศของผู้เข้าประกวดดังกล่าว โดย 1 แทนผู้ชายและ 2 แทนผู้หญิง

จงเขียนโปรแกรมที่รับค่า K, K_1, K_2, S_1, S_2 และ G เข้ามาตามลำดับ และโปรแกรมจะทำการพิจารณาว่า ผู้เข้าประกวดที่โปรแกรมกำลังพิจารณา จะได้อยู่ในการดูแลของกรรมการท่านใด หากได้อยู่กับกรรมการคนที่ 1 โปรแกรมก็จะพิมพ์เลข 1 มาเป็นผลลัพธ์ หากเป็นกรรมการคนที่ 2 โปรแกรมก็จะพิมพ์เลข 2 ออกมา แต่หากไม่มีกรรมการท่านใดรับผู้เข้าประกวดไปดูแล โปรแกรมก็จะพิมพ์เลข 0 ออกมา

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
10 3 5 9 9 1	1
10 3 5 9 9 2	2
10 10 5 9 9 1	2
10 3 10 9 9 2	1
10 3 5 8 8 1	0

ปัญหา 19 รับคำสั่งซื้อลูกค้าได้หรือไม่ [consider_order]

โรงงานเฟอร์นิเจอร์แห่งหนึ่งรับผลิตโต๊ะสองขนาดคือขนาดใหญ่และขนาดเล็ก โต๊ะขนาดใหญ่ใช้ไม้แผ่น 2 แผ่น และไม้ท่อน 6 ท่อน ส่วนโต๊ะขนาดเล็กใช้ไม้แผ่น 1 แผ่น และไม้ท่อน 4 ท่อน ณ ขณะนี้โรงงานมีไม้แผ่นและไม้ท่อนเป็นจำนวน X และ Y ตามลำดับ นอกจากนี้ลูกค้าต้องการสั่งซื้อโต๊ะใหญ่และเล็กเป็นจำนวน M และ N โต๊ะตามลำดับ

จงเขียนโปรแกรมที่คำนวณว่าจะรับคำสั่งซื้อของลูกค้าได้หรือไม่ ทั้งนี้หากวัสดุมีเพียงพอที่จะผลิตโต๊ะได้ครบตามที่ลูกค้าสั่งทั้งหมด จะถือว่าสามารถรับคำสั่งซื้อได้และโปรแกรมจะพิมพ์คำว่า Yes พร้อมทั้งแสดงจำนวนไม้แผ่นและไม้ท่อนที่เหลือจากการผลิตตามคำสั่งซื้อ แต่หากวัสดุไม่พอก็จะไม่สามารถรับคำสั่งซื้อของลูกค้าได้ โปรแกรมจะพิมพ์คำว่า No พร้อมทั้งแสดงจำนวนไม้แผ่นและไม้ท่อนที่โรงงานต้องจัดหาเพิ่มเพื่อให้ผลิตโต๊ะตามยอดสั่งซื้อของลูกค้าได้พอดี

ข้อมูลเข้า

- บรรทัดที่หนึ่งคือค่า X และ Y ตามลำดับ เป็นจำนวนเต็มที่ไม่น้อยกว่าศูนย์ คั่นด้วยช่องว่าง
- บรรทัดที่สองคือค่า M และ N ตามลำดับ เป็นจำนวนเต็มที่ไม่น้อยกว่าศูนย์ คั่นด้วยช่องว่าง นอกจากนี้ M และ N จะไม่เป็นศูนย์พร้อมกัน

ผลลัพธ์

มีหนึ่งบรรทัดคือคำว่า Yes หรือ No ตามด้วยช่องว่างจากนั้น

- ถ้าผลลัพธ์เป็น Yes ให้ระบุจำนวนไม้แผ่นและไม้ท่อนที่เหลือจากการผลิตตามลำดับ จำนวนทั้งสองคั่นด้วยช่องว่างหนึ่งช่อง
- ถ้าผลลัพธ์เป็น No ให้ระบุจำนวนไม้แผ่นและไม้ท่อนที่โรงงานต้องจัดหาเพิ่มเพื่อผลิตโต๊ะตามยอดสั่งซื้อได้พอดี

[อธิบายเพิ่มเติม คำว่าพอดีก็คือ การจัดหาวัสดุมาเพิ่มให้ให้น้อยที่สุด นั่นคือถ้าหากขาดไม้แผ่น 5 แผ่นและไม้ท่อน 10 ท่อน โรงงานก็ต้องจัดหาไม้มาเพิ่ม 5 แผ่นและ 10 ท่อนตามลำดับ แต่ถ้าไม้แผ่นมีเกินอยู่ 8 แผ่นขาดแต่ไม้ท่อน 10 ท่อน โรงงานก็ต้องหาไม้เพิ่ม 0 แผ่นและ 10 ท่อน ในกรณีนี้เราจะพบว่าสุดท้ายก็จะเหลือไม้แผ่นในการผลิตอยู่บ้าง และไม่จำเป็นที่วัสดุทุกอย่างจะถูกใช้หมดไปอย่างสิ้นเชิง]

ตัวอย่าง

ตัวอย่างที่ 1		ตัวอย่างที่ 2	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
100 200 5 0	Yes 90 170	30 100 10 20	No 10 40
ตัวอย่างที่ 3		ตัวอย่างที่ 4	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
100 400 40 20	Yes 0 80	100 5 5 0	No 0 25
ตัวอย่างที่ 5		ตัวอย่างที่ 6	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
120 320 40 20	Yes 20 0	80 400 40 20	No 20 0

[หมายเหตุ ในตัวอย่างที่สี่ จำนวนไม้แผ่นที่โรงงานมีอยู่นั้นมีมากเกินไปที่จะผลิตโต๊ะได้ ขาดแต่เพียงไม้ท่อน ดังนั้นจำนวนไม้แผ่นที่โรงงานต้องหามาเพิ่มจึงเป็น 0 แต่ของไม้ท่อนต้องหามาเพิ่ม 25]

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่หก

การวนซ้ำและการจัดรูปแบบการแสดงผล

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

คำสั่ง

1. ให้เขียนโปรแกรมภาษาซีสำหรับปัญหาที่ให้ไป
2. ระเบียบการส่งงานเขียนโปรแกรมก็คือนักศึกษาจะส่งโค้ดเข้าไปโปรแกรมตรวจงาน

ปัญหาชุดที่ 1 หัดใช้โครงสร้างการวนซ้ำ

จุดประสงค์การเรียนรู้ 1: การวนซ้ำ (loop) เป็นพื้นฐานที่สำคัญในการแก้ปัญหาด้านการคำนวณ อย่างไรก็ตาม โครงสร้างของการวนซ้ำนั้นซับซ้อนกว่าโครงสร้างเงื่อนไขทั่วไป ทั้งในแง่ของแนวคิด และ ไวยากรณ์ของภาษาโปรแกรม ในจุดประสงค์การเรียนรู้นี้ ผู้เรียนจะได้ทำความเข้าใจกับโครงสร้างการวนซ้ำ เพื่อเป็นการเตรียมตัวสำหรับการแก้ปัญหาที่มีซับซ้อนมากขึ้น

โครงสร้างการวนซ้ำที่เราจะนำมาใช้เป็นตัวอย่งสำหรับการเรียนนี้มีอยู่สองแบบคือ แบบที่ใช้ while และแบบที่ใช้ for ในแบบฝึกหัดข้อแรก เราจะศึกษาตัวอย่างโค้ดสำหรับงานนี้ไปด้วยกัน

ปัญหา 1 วนไปพิมพ์ไป [print_loop]

จงเขียนโปรแกรมที่พิมพ์เลขจำนวนเต็มจาก 0 ถึง 19 ออกมา (รวมเลข 0 และ 19 ด้วย) หนึ่งตัวต่อหนึ่งบรรทัด

วิธีทำ

```
#include <stdio.h>

void main() {
    int i = 0;
    while(i < 20) {
        printf("%d\n", i);
        i += 1;
    }
}
```

โปรแกรมนี้นั้นสั้นมาก แต่บรรยายโครงสร้างของการวนซ้ำทั้ง 4 ประการครบ นั่นก็คือ (1) การเตรียมตัวก่อนเข้าลูป ซึ่งก็คือการตั้งค่าตัวแปร [int i = 0;], (2) โครงสร้างลูปพร้อมเงื่อนไขลูป [while(i < 20) {...}], (3) งานที่จะให้ทำซ้ำ ซึ่งก็คือการพิมพ์ค่าตัวเลขหนึ่งตัวต่อบรรทัด [printf("%d\n", i);] และ (4) การเปลี่ยนตัวแปรเงื่อนไขลูป [i += 1;]

ในปัญหาข้อต่อไป เราจะลองเปลี่ยนเงื่อนไขลูปและค่าเริ่มต้นของตัวแปรสอดคล้องกับปัญหาและข้อมูลเข้าจากผู้ใ้

ปัญหา 2 วนไปพิมพ์ไปภาคสอง [print_loop2]

จงเขียนโปรแกรมที่พิมพ์เลขจำนวนเต็มจาก 1 ถึง N ออกมา (รวมเลข 1 และ N ด้วย) หนึ่งตัวต่อหนึ่งบรรทัด โดย N เป็นจำนวนเต็มบวกที่รับมาจากผู้ใช้ แต่ในกรณีที่ผู้ใช้ใส่เลข 0 หรือเลขติดลบ โปรแกรมจะพิมพ์คำว่า Invalid input และจบการทำงานทันที

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
4	1 2 3 4	1	1	0	Invalid input

ปัญหา 3 วนไปในขอบเขตที่กำหนด [loop_range]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็ม x และ y มาจากผู้ใช้ โดยที่ $x \leq y$ โปรแกรมจะพิมพ์เลขจำนวนเต็มจาก x ถึง y ออกมา (รวมเลข x และ y ด้วย) การพิมพ์ตัวเลขเหล่านี้จะพิมพ์แต่ละตัวแล้วคั่นด้วยช่องว่างหนึ่งช่องแล้วจึงพิมพ์ตัวถัดไป สำหรับกรณีที่ $x > y$ โปรแกรมจะพิมพ์คำว่า Invalid input ออกมา

วิธีพิมพ์ชุดตัวเลขแบบมีช่องว่างคั่นสามารถทำได้โดยการพิมพ์เลขแต่ละตัวและพิมพ์ช่องว่างตามมาด้วยเสมอหนึ่งช่องทันที เมื่อมีตัวเลขตามมา ตัวเลขจะอยู่หลังช่องว่างดังกล่าวโดยปริยายทำให้ตัวเลขถูกคั่นด้วยช่องว่างหนึ่งช่องตามที่โจทย์ระบุ กล่าวคือถ้าเราใช้คำสั่ง `printf("%d ", k);` การพิมพ์ตัวเลขจะมีช่องว่างตามมาด้วยทันทีตามที่แนะนำ [สังเกตให้ดีว่ามีช่องว่างตาม %d มาก่อนที่จะปิดอัญประกาศคู่]

ข้อมูลเข้า	ผลลัพธ์
2 5	2 3 4 5
-3 8	-3 -2 -1 0 1 2 3 4 5 6 7 8
5 2	Invalid input

ปัญหา 4 วนไปในขอบเขตที่กำหนดภาคสอง [loop_range2]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็ม x และ y มาจากผู้ใช้ โปรแกรมจะพิมพ์เลขจำนวนเต็มจาก x ถึง y ออกมา (รวมเลข x และ y ด้วย) โดยที่ถ้า $x > y$ ก็จะพิมพ์เลขแบบนับถอยหลัง แต่ถ้า $x \leq y$ ก็จะพิมพ์แบบนับเดินหน้า

คำแนะนำ ในแบบฝึกหัดข้อนี้ ผู้เรียนจะได้เริ่มเห็นว่า เราสามารถนำโครงสร้างการวนซ้ำไปใส่ไว้ในเงื่อนไข `if` ได้ และในโปรแกรมอันหนึ่งจะมีโครงสร้างการวนซ้ำหลายตำแหน่งก็ได้

ปัญหาชุดที่ 2 เทคนิคพื้นฐาน

จุดประสงค์การเรียนรู้ 2: ในแบบฝึกหัดชุดที่แล้ว โปรแกรมของเรามีการรับข้อมูลเข้าจากผู้ใช้เข้ามาเพียง 2 ค่าเป็นอย่างมากเท่านั้น อย่างไรก็ตาม ปัญหาการคำนวณจำนวนมากมีการรับข้อมูลเข้าจากผู้ใช้เป็นปริมาณมาก ในแบบฝึกหัดนี้เราจะสร้างความคุ้นเคยในการรับข้อมูลเข้าหลายจำนวนด้วยการวนลูป

นอกจากนี้เราจะเรียนรู้เทคนิคที่พบบ่อย เช่น การหาค่าสูงสุดต่ำสุด และการเปรียบเทียบค่าเพื่อหาค่าที่ใกล้เคียงที่สุด ลูปในแบบฝึกหัดนี้อาจจะมีจำนวนการวนรอบที่น้อย และมีจำนวนที่แน่นอนเพื่อให้ปัญหาคงง่ายขึ้น และทำให้ผู้เรียนสามารถผสมเงื่อนไขการพิจารณาของ if ที่ซับซ้อนขึ้นกว่าปัญหาชุดที่แล้วกับแนวคิดการวนรับข้อมูลเข้าในลูปได้ ความน้อยของจำนวนรอบลูปทำให้บางคนอาจจะเลือกการคัดลอกโค้ดซ้ำ ๆ กันหลายชุด แล้วไปแก้แต่ละจุด วิธีนั้นถึงแม้จะได้คำตอบที่ถูกต้องแต่ก็ไม่เป็นการฝึกใช้ลูปในรูปแบบที่ควรจะเป็น ดังนั้นขอให้ผู้เรียนเลือกใช้ลูปกับปัญหาทุกข้อในแบบฝึกหัดสัปดาห์นี้

ปัญหา 5 จำนวนบวกที่น้อยและมากที่สุด [positive_min_max]

จงเขียนโปรแกรมที่รับค่าข้อมูลเข้าจากผู้ใช้เป็นตัวเลขจำนวนเต็ม 8 ค่า ตัวเลขนี้มีได้ทั้งค่าบวกค่าลบและเลขศูนย์ อย่างไรก็ตามในปัญหานี้ตัวเลขค่าบวกเป็นที่สนใจเป็นพิเศษ จึงมีความพยายามที่จะตอบคำถามให้ได้ว่าในบรรดาเลขที่ผู้ใช้ใส่เข้ามาทั้งแปดค่านั้น

1. เลขบวกที่มีค่ามากที่สุดมีค่าเท่าใด
2. เลขบวกที่มีค่าน้อยที่สุดมีค่าเท่าใด

ข้อมูลเข้า

เป็นตัวเลขจำนวนเต็มแปดค่า รับเข้าในโปรแกรมทาง standard input (โดยทั่วไปแล้วก็คือคีย์บอร์ดของเครื่องคอมพิวเตอร์นั่นเอง) การใส่ข้อมูลเข้าจะใส่เข้ามาหนึ่งค่าต่อหนึ่งบรรทัด (ดังนั้นข้อมูลเข้าจะมี 8 บรรทัดเพราะมีเลข 8 ค่าจากผู้ใช้) **หมายเหตุ** ข้อมูลเข้าในปัญหานี้จะมีเลขบวกอย่างน้อยหนึ่งค่าเสมอ

ผลลัพธ์

1. บรรทัดแรกระบุเลขบวกที่มีค่ามากที่สุดที่พบในข้อมูลเข้า
2. บรรทัดที่สองระบุเลขบวกที่มีค่าน้อยที่สุดที่พบในข้อมูลเข้า

[มีตัวอย่างและคำแนะนำสำหรับผู้เริ่มต้นในหน้าถัดไป]

ตัวอย่าง

ตัวอย่างที่ 1		ตัวอย่างที่ 2	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
7	20	-5	8
3	2	3	3
-1		7	
0		0	
-10		-5	
2		8	
5		-100	
20		8	
ตัวอย่างที่ 3		ตัวอย่างที่ 4	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
0	7	5	100
0	7	-3	5
7		-7	
0		0	
-1		5	
-5		-8	
-2		100	
-7		-8	

คำแนะนำ

วิธีแก้ปัญหามีหลายแบบแต่วิธีที่แนะนำต่อไปนี้เป็นหนึ่งในวิธีที่ง่ายที่สุด

1. ควรจะมีการใช้รูปเพื่อรับค่าและเปรียบเทียบค่าความมากน้อยของตัวเลขที่ผู้ใช้ใส่เข้ามา
2. การเปรียบเทียบค่าจะต้องแยกให้ออกด้วยว่าจากเลขที่ผู้ใช้ใส่เข้ามา ตัวไหนเป็นเลขบวก เราจะนำมาเปรียบเทียบความมากน้อยเฉพาะเลขบวก ส่วนข้อมูลเข้าที่ไม่ใช่เลขบวกก็ไม่ต้องนำมาเปรียบเทียบค่า
3. ควรจะมีตัวแปรคอยบันทึกค่าบวกที่มากที่สุดและที่น้อยที่สุดไว้ด้วย
4. หลังจากวนรูปเพื่อรับค่าและเปรียบเทียบตัวเลขแล้ว จึงแสดงผลลัพธ์ออกมาเป็นคำตอบในคราวเดียว
5. ปัญหาข้อนี้ไม่จำเป็นต้องใช้อาเรย์หรือสตรัค แต่ถ้านักศึกษาชอบใช้อาเรย์หรือสตรัค ก็สามารถใช่วิธีดังกล่าวได้ เพราะแท้จริงแล้วปัญหาข้อนี้มีวิธีทำหลายแบบ

ปัญหา 6 คู่หรือคือใครแน่จริง [odd_vs_even]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็มจากผู้ใช้เป็นจำนวนทั้งหมด 8 ค่า โปรแกรมนี้จะต้องทำการหาผลบวกของเลขคู่ทั้งหมดและหาผลบวกของเลขคี่ทั้งหมดที่อยู่ในจำนวนเต็มทั้ง 8 ค่านั้น สุดท้ายโปรแกรมจะต้องระบุว่าระหว่างผลบวกของเลขคู่กับเลขคี่ ไหนที่มีค่ามากกว่ากัน พร้อมทั้งแสดงผลบวกของเลขทั้งสองกลุ่มด้วย

ข้อมูลเข้า

เป็นตัวเลขจำนวนเต็มแปดค่า รับเข้าในโปรแกรมทาง standard input (โดยทั่วไปแล้วก็คือคีย์บอร์ดของเครื่องคอมพิวเตอร์นั่นเอง) การใส่ข้อมูลเข้าจะใส่เข้ามาหนึ่งค่าต่อหนึ่งบรรทัด (ดังนั้นข้อมูลเข้าจะมี 8 บรรทัดเพราะมีเลข 8 ค่าจากผู้ใช้งาน) **หมายเหตุ** ข้อมูลเข้าในปัญหานี้จะมีเลขคู่และเลขคี่อยู่อย่างน้อยหนึ่งค่าเสมอ

ผลลัพธ์

โปรแกรมจะต้องแสดงผลลัพธ์ออกมาสามอย่างดังนี้

- บรรทัดแรกระบุว่า จากข้อมูลเข้า ผลบวกเลขคู่หรือคืออย่างไรมีค่ามากกว่ากัน หากเป็นผลบวกเลขคู่ให้แสดงว่า even ถ้าเป็นเลขคี่ให้พิมพ์ว่า odd แต่ถ้าเท่ากันให้พิมพ์ว่า equal
- บรรทัดที่สองแสดงผลบวกของเลขคู่ในข้อมูลเข้า
- บรรทัดที่สามแสดงผลบวกของเลขคี่ในข้อมูลเข้า

ตัวอย่าง

ตัวอย่างที่ 1		ตัวอย่างที่ 2	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
7 3 -1 0 -10 2 5 20	odd 12 14	-5 3 7 0 -5 8 -100 6	odd -86 0
ตัวอย่างที่ 3		ตัวอย่างที่ 4	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
0 0 7 0 3 -5 -2 -7	equal -2 -2	2 3 -7 0 9 -8 100 -8	even 86 5

ปัญหา 7 หาจำนวนที่มีค่าใกล้เคียงที่สุดกับเลขที่กำหนด [ClosestNumber]

จงเขียนโปรแกรมที่รับค่าข้อมูลเข้าจากผู้ใช้ โดยตัวเลขตัวแรกเป็นจำนวนเต็ม X ต่อจากนั้นโปรแกรมจะรับตัวเลขจำนวนเต็มอีก 8 ค่า จงเขียนโปรแกรมเพื่อหาว่า จากเลขทั้ง 8 ตัวนั้นเลขใดที่ใกล้เคียงกับ X มากที่สุด

ข้อมูลเข้า

- บรรทัดแรกเป็นเลขจำนวนเต็ม X
- บรรทัดที่สองถึงเก้า เลขจำนวนเต็มอีกแปดค่าที่ต้องการทดสอบว่าเลขใดที่ใกล้เคียงกับ X มากที่สุด

ข้อมูลเข้าทั้งหมดรับเข้าในโปรแกรมทาง standard input (โดยทั่วไปแล้วก็คือคีย์บอร์ดของเครื่องคอมพิวเตอร์นั่นเอง) การใส่ข้อมูลเข้าจะใส่เข้ามาหนึ่งค่าต่อหนึ่งบรรทัด (ดังนั้นข้อมูลเข้าจะมี 9 บรรทัดเพราะมีค่า X และเลขอีก 8 ค่าจากผู้ใช้)

หมายเหตุ เพื่อทำปัญหาให้ง่าย ปัญหานี้ถูกกำหนดให้ตัวเลขที่จะมีค่าใกล้เคียงกับ X มากที่สุดมีเพียงตัวเดียวเท่านั้น

หมายเหตุ เลขข้อมูลเข้าเป็นจำนวนเต็มบวกหรือศูนย์ มีค่าไม่เกิน 500 ล้าน

ผลลัพธ์

ค่าของเลขจำนวนเต็มจากข้อมูลเข้าแปดตัวหลังสุดที่ใกล้เคียงกับ X มากที่สุด

[มีตัวอย่างและคำแนะนำสำหรับผู้เริ่มต้นในหน้าถัดไป]

ตัวอย่าง

ตัวอย่างที่ 1		ตัวอย่างที่ 2	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
-7	-10	100	8
7		-5	
3		3	
-1		7	
0		0	
-10		-5	
2		8	
5		-100	
20		6	

ตัวอย่างที่ 3		ตัวอย่างที่ 4	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
-7	-7	90	100
0		5	
0		-3	
7		-7	
0		0	
-1		5	
-5		-8	
-2		100	
-7		-8	

คำแนะนำ

วิธีแก้ปัญหามีหลายแบบแต่วิธีที่แนะนำต่อไปนี้เป็นหนึ่งในวิธีที่ง่ายที่สุด

1. ควรจะมีการใช้รูปเพื่อรับค่าและเปรียบเทียบค่าความแตกต่างระหว่าง X กับค่าอื่น ๆ ที่รับเข้ามา
2. ควรจะมีตัวแปรคอยบันทึกทั้งค่าความแตกต่างที่น้อยที่สุดและตัวเลขที่อาจจะเป็นคำตอบสุดท้าย
3. หลังจากวนลูปเพื่อรับค่าและเปรียบเทียบตัวเลขแล้ว จึงแสดงผลลัพธ์ออกมาเป็นคำตอบ.
4. ปัญหาข้อนี้ไม่จำเป็นต้องใช้อาเรย์หรือสตรัค แต่ถ้านักศึกษาชอบใช้อาเรย์หรือสตรัค ก็สามารถใช่วิธีดังกล่าวได้ เพราะแท้จริงแล้วปัญหาข้อนี้มีวิธีทำหลายแบบ

ปัญหาชุดที่ 3 การนับจำนวน

จุดประสงค์การเรียนรู้ 3: ในปัญหาชุดที่แล้ว จำนวนรอบในการวนลูปมีจำนวนที่แน่นอน อย่างไรก็ตาม ปัญหาการคำนวณหลายอย่าง โปรแกรมอาจไม่ทราบจำนวนข้อมูลเข้าล่วงหน้าและจำนวนรอบการวนลูปจึงไม่แน่นอนตายตัว แบบฝึกหัดชุดนี้จะสร้างความคุ้นเคยในการรับข้อมูลเข้าจากผู้ใช้โดยการวนซ้ำรับข้อมูลเข้าปริมาณมาก พร้อมกับเรียนรู้เทคนิคการนับจำนวนที่หลากหลายไปในตัว

ปัญหา 8 หยุตลูปโดยพลัน [how_to_break]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็มจากผู้ใช้เข้ามาจนกว่าจะพบเลขศูนย์ จากนั้นให้พิมพ์จำนวนตัวเลขจากผู้ใช้ที่ไม่เป็นศูนย์ออกมาเป็นผลลัพธ์

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
5	4	0	0
8			
-1			
1			
0			
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
-1	5	57	1
7			
3			
9			
-100			
0			

ปัญหา 9 นับจำนวนที่สนใจ [count_target_number]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็มจากผู้ใ้ โดยจำนวนแรกคือเลขที่ผู้ใช้สนใจ จากนั้นโปรแกรมจะรับเลขจำนวนเต็มเข้ามาจนกว่าจะพบเลขศูนย์ (หมายถึงเลขศูนย์โดด ๆ ไม่ใช่เลขศูนย์ที่อยู่กับเลขตัวอื่น เช่น 10) เมื่อรับข้อมูลเข้าจนครบแล้ว โปรแกรมจะรายงานว่าพบตัวเลขที่ผู้ใช้สนใจทั้งหมดกี่จำนวน แต่ถ้าไม่พบเลขที่ผู้ใช้สนใจเลย โปรแกรมจะพิมพ์คำว่า None

หมายเหตุ 1 เลขที่ผู้ใช้สนใจจะไม่เป็นเลขศูนย์อย่างแน่นอน

หมายเหตุ 2 การพิมพ์คำว่า None ออกมา ตัว N ตัวแรกเป็นตัวพิมพ์ใหญ่ ส่วนตัวอื่น ๆ ต้องเป็นตัวพิมพ์เล็ก

ตัวอย่าง

ตัวอย่างที่ 1		ตัวอย่างที่ 2		ตัวอย่างที่ 3	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
5	6	-2	5	5	None
2		7		2	
-5		-2		-2	
5		100		7	
5		-2		8	
5		-2		9	
3		-2		10	
5		-2		0	
8		5			
5		0			
5					
0					

ปัญหา 10 นับจำนวนเลขซ้ำติดกันที่เราสนใจ [target_max_consec]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็มจากผู้ใช้ โดยจำนวนแรกคือเลขที่ผู้ใช้สนใจ จากนั้นโปรแกรมจะรับเลขจำนวนเต็มเข้ามาจนกว่าจะเจอเลขศูนย์ เมื่อรับข้อมูลเข้าจนครบแล้วโปรแกรมจะรายงานว่าตัวเลขที่ผู้ใช้สนใจถูกใส่เข้ามาติดต่อกันโดยไม่มีเลขตัวอื่นมาคั่นยาวที่สุดกี่จำนวน **หมายเหตุ** เลขที่ผู้ใช้สนใจจะไม่เป็นเลขศูนย์และการนับจำนวนเลขที่ติดกันนี้ไม่นับเลขตัวแรกที่เป็นตัวระบุว่าคุณสนใจเลขค่าใด (ดูตัวอย่างที่ 4)

ข้อมูลเข้า

- บรรทัดแรกระบุตัวเลขที่ผู้ใช้สนใจ ซึ่งเป็นจำนวนเต็มที่มีค่าไม่เท่ากับศูนย์
- บรรทัดที่เหลือเป็นเลขจำนวนเต็ม หนึ่งตัวหนึ่งบรรทัด โดยบรรทัดสุดท้ายเป็นเลขศูนย์

ผลลัพธ์

- บรรทัดแรกเป็นเลขจำนวนเต็มระบุว่าคุณสนใจเลขซ้ำติดต่อกันมากที่สุดโดยไม่มีเลขอื่นมาคั่นกี่จำนวน
- บรรทัดที่สองเป็นตัวเลขจำนวนเต็มระบุว่าคุณสนใจเลขทั้งหมดกี่จำนวน

หมายเหตุ การนับจำนวนจะไม่รวมตัวเลขที่ใส่ระบุค่าที่ผู้ใช้สนใจ (ค่าในข้อมูลเข้าบรรทัดแรก)

ตัวอย่าง

ตัวอย่างที่ 1		ตัวอย่างที่ 2		ตัวอย่างที่ 3		ตัวอย่างที่ 4	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
5	3	-2	4	5	0	7	2
2	6	7	5	2	0	7	3
-5		-2		-2		7	
5		100		7		2	
5		-2		8		7	
5		-2		9		0	
3		-2		10			
5		-2		0			
8		5					
5		0					
5							
0							

อธิบายตัวอย่าง จากตัวอย่างที่ 1 ผู้ใช้ใส่ตัวเลข 5 เข้ามาติดต่อกันสูงสุด 3 ตัว ผลลัพธ์จึงมีค่าเท่ากับ 3 ส่วนตัวอย่างที่ 2 ผู้ใช้ใส่เลข -2 เข้ามาติดต่อกันสูงสุด 4 ตัว สำหรับตัวอย่างที่ 4 เนื่องจากเลข 7 ตัวแรกใช้ในการระบุค่าที่ผู้ใช้สนใจ จึงไม่นับมารวมเป็นเลขที่ติดกัน คำตอบจึงมีค่าเท่ากับ 2

ปัญหา 11 นับจำนวนเลขซ้ำติดกันที่ยาวที่สุด [any_max_consec]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็มจากผู้ใช้งานว่าจะพบเลขศูนย์ เมื่อรับข้อมูลเข้าจนครบแล้ว โปรแกรมจะรายงาน ว่าเลขที่ถูกใส่เข้ามาซ้ำติดต่อกันโดยไม่มีเลขตัวอื่นคั่น ติดต่อกันที่ยาวที่สุดกี่ครั้งและเลขดังกล่าวมีค่าเท่าใด สำหรับ โปรแกรมนี้ ผลลัพธ์ในบรรทัดแรกจะระบุจำนวนตัวเลขที่ซ้ำติดต่อกันมากที่สุด ส่วนบรรทัดที่สองระบุว่าเลขที่ซ้ำติดต่อกันมากที่สุดคือเลขตัวใด ในกรณีที่มีเลขที่ซ้ำติดต่อกันยาวที่สุดหลายจำนวน ให้รายงานเฉพาะเลขตัวแรกที่ซ้ำติดต่อกันมากที่สุด (ดูตัวอย่างที่ 3 และ 4)

คำแนะนำ ข้อนี้คล้ายกับข้อ target_max_consec แต่กลไกการคิดถือว่าซับซ้อนขึ้นพอสมควรสำหรับผู้เริ่มต้น ตัวอย่าง

ตัวอย่างที่ 1		ตัวอย่างที่ 2		ตัวอย่างที่ 3		ตัวอย่างที่ 4	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
2	3	-2	4	5	1	2	3
-5	5	7	-2	2	5	7	7
5		-2		-2		7	
5		100		7		7	
5		-2		8		2	
3		-2		9		2	
5		-2		10		1	
8		-2		0		2	
5		5				2	
5		0				2	
0						0	

ในตัวอย่างที่ 3 เลขแต่ละตัวต่างก็ถูกใส่เข้ามาติดต่อกันเพียงครั้งเดียว ดังนั้นโปรแกรมจึงรายงานเฉพาะเลขตัวแรก ซึ่งก็คือเลข 5 ส่วนตัวอย่างที่ 4 เราจะเห็นว่าเลข 7 และเลข 2 ต่างก็ซ้ำติดต่อกันสูงสุดสามครั้ง แต่เลข 7 ซ้ำติดต่อกันมากที่สุดก่อนเลข 2 โปรแกรมจึงรายงานเลข 7 ออกมาในผลลัพธ์

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่เจ็ด

การวนซ้ำและการจัดรูปแบบการแสดงผล 2

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

คำสั่ง

1. ให้เขียนโปรแกรมภาษาซีสำหรับปัญหาที่ให้ไป
2. ระเบียบการส่งงานเขียนโปรแกรมก็คือนักศึกษาจะส่งโค้ดเข้าไปโปรแกรมตรวจงาน

จุดประสงค์การเรียนรู้: ในสัปดาห์นี้เราจะเรียนรู้การใช้ลูปกับการแก้ปัญหาที่จำลองมาจากเหตุการณ์จริงมากขึ้น เพื่อเสริมทักษะการเขียนโปรแกรมและสร้างพื้นฐานการแก้ปัญหาเชิงคำนวณที่ซับซ้อนขึ้นได้

ปัญหา 1 วางแผนผลิตชาเขียว [green_tea]

โรงงานผลิตชาเขียวใช้น้ำ 250 ลูกบาศก์เซนติเมตร และ น้ำตาล 15 กรัมต่อชาเขียวหนึ่งขวด ผู้คุมโรงงานวางแผนการผลิตโดยการระบุปริมาณน้ำและน้ำตาลที่โรงงานมีในแต่ละวันให้กับโปรแกรม โดยค่าที่ป้อนให้โปรแกรมนี้เป็นเลขจำนวนเต็ม โดยข้อมูลเข้าคือปริมาณน้ำและน้ำตาลตามลำดับ โปรแกรมจะวางแผนการผลิตโดยคำนวณจำนวนขวดชาเขียวที่ผลิตได้สูงสุดจากวัตถุดิบในวันนั้นพร้อมทั้งแสดงผลออกมาทางหน้าจอ นอกจากนี้โปรแกรมจะพิมพ์ข้อความว่า 'water' หากมีน้ำเหลือจากการผลิตในวันดังกล่าว จงเขียนโปรแกรมภาษาซี โดยกำหนดให้ผู้คุมโรงงานใส่ข้อมูลวางแผนการผลิตเป็นจำนวน 7 วัน (นั่นคือมีการวนทำซ้ำรับข้อมูลและแสดงผลเป็นจำนวน 7 รอบ)

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
500 30	2
500 31	2
501 30	2 water
501 32	2 water
1000 500	4
2000 1	0 water
0 50	0

ปัญหา 2 วันที่ตัดเสื้อแล้วเสร็จ [due_date]

โรงงานตัดเย็บเสื้อแห่งหนึ่ง รับคำสั่งตัดเสื้อจากลูกค้ามาทั้งหมด N คำสั่ง โดยในแต่ละคำสั่งจะเป็นปริมาณเสื้อที่ลูกค้าต้องการ กำหนดให้โรงงานมีพนักงานทั้งหมด K คนและพนักงานหนึ่งคนจะผลิตเสื้อได้ 100 ตัวต่อวัน นอกจากนี้โรงงานจะทำการผลิตเสื้อจากคำสั่งตัดเสื้อคำสั่งแรกเสร็จแล้วจึงทำคำสั่งตัดที่สอง สาม เรียงตามลำดับเช่นนี้ไปตลอด เนื่องจากการคำนวณเวลาที่งานจะแล้วเสร็จเป็นสิ่งสำคัญในการประมาณวันรับสินค้าของลูกค้า โรงงานจึงพยายามทำการคำนวณเวลาที่ตัดเสื้อของแต่ละคำสั่งแล้วเสร็จ

จงเขียนโปรแกรมหาว่าโรงงานจะตัดเย็บเสื้อผ้าตามคำสั่งซื้อแต่ละอันเสร็จในวันที่เท่าใดนับจากวันแรก (วันแรกคือวันที่หนึ่ง)

หมายเหตุ ถ้าคนงานตัดเสื้อของคำสั่งซื้อคำสั่งที่หนึ่งเสร็จ และยังมีเหลือเวลา ก็จะทำการตัดเสื้อของคำสั่งตัดเสื้อที่ตามมาเรื่อย ๆ จนกว่าคนงานคนดังกล่าวจะตัดเสื้อครบร้อยตัวในวันนั้น หรือคำสั่งตัดเสื้อทุกอันถูกทำงานเสร็จทุกงานแล้ว เช่น หากคนงาน 2 คน และมี 4 คำสั่งตัดเสื้อจากลูกค้า เรียงตามลำดับดังนี้

90

60

70

40

ความสามารถในการตัดเสื้อต่อวันของโรงงานคือ $100 \times 2 = 200$ ตัวต่อวัน (คนงานสองคน) หลังจากทำคำสั่งแรกเสร็จซึ่งคิดรวมเป็นเสื้อได้ 90 ตัว คนงานจะทำคำสั่งที่สองต่อไปในวันเดียวกัน ซึ่งทำให้ผลิตเสื้อออกมารวมได้ 150 ตัว จากสองคำสั่งซื้อ คนงานจึงตัดเสื้อตามคำสั่งตัดที่สามต่อ แต่จะตัดได้แค่ 50 ตัว ที่เหลืออีก 20 ตัวในคำสั่งตัดที่สามจะต้องมาทำต่อในวันถัดไป พอถึงวันถัดมาคนงานก็ตัดเย็บเสื้อ 20 ตัวนั้นจนเสร็จและตัดเสื้อตามคำสั่งซื้อสุดท้ายเสร็จในวันเดียวกัน

ข้อมูลเข้า

1. บรรทัดแรกระบุจำนวนคนงาน (ค่า K) เป็นจำนวนเต็มบวก โดยที่ K ไม่เกิน 1,000
2. บรรทัดที่สองระบุจำนวนคำสั่งตัดเสื้อ (ค่า N) เป็นจำนวนเต็มบวก โดยที่ N มีค่าไม่เกิน 10,000
3. อีก N บรรทัดต่อมา ระบุจำนวนเสื้อที่สั่งตัดในแต่ละคำสั่งตัด หนึ่งคำสั่งต่อหนึ่งบรรทัด เรียงจากคำสั่งแรกไปคำสั่งสุดท้าย โดยที่จำนวนเสื้อในแต่ละคำสั่งจะไม่เกิน 10,000

ผลลัพธ์

มี N บรรทัดเรียงตามลำดับคำสั่งตัดเสื้อคือ ในแต่ละบรรทัดระบุว่าคำสั่งตัดเสื้อแต่ละคำสั่งจะเสร็จในวันที่เท่าใด (ให้นับจำนวนวันเริ่มจาก 1 ไปเรื่อย ๆ) บรรทัดแรกแทนวันที่ตัดเสื้อของคำสั่งซื้อแรกแล้วเสร็จ

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
2	1	2	2	5	
4	1	4	3	10	
90	2	210	3	400	1
60	2	200	4	400	2
70		190		500	3
40		200		1700	6
				5000	16
				1300	19
				750	21
				280	21
				375	22
				2000	26

ปัญหา 3 จองรถตู้ [reserve_3_vans]

บริษัทแห่งหนึ่งมีรถตู้อยู่ 3 คันที่พนักงานสามารถนำไปใช้ได้ รถตู้คันที่หนึ่ง สอง และ สาม มีรหัสประจำรถว่า A, B และ C ตามลำดับ ข้อกำหนดในการนำไปใช้มีอยู่ว่าพนักงานจะต้องทำการจองรถก่อน โดยคำสั่งจองจะต้องระบุจำนวนวันที่จะใช้ จากนั้นผู้จองจะได้รถตู้ที่ว่างให้ใช้เร็วที่สุดเท่าที่หาได้จากหนึ่งในสามคันนั้น

ในกรณีที่รถตู้ว่างให้ใช้เร็วที่สุดมากกว่าหนึ่งคันและ A ว่างให้ใช้เร็วที่สุด A จะถูกเลือกก่อน B และ C (เป็นได้ว่าจะว่างให้ใช้เร็วที่สุดพร้อมกันทั้งสามคัน หรือแค่สองคันซึ่งเป็น A กับ B หรือ A กับ C ก็ได้) ถ้า A ไม่ได้ว่างให้ใช้เร็วที่สุด แต่เป็น B กับ C ที่ว่างให้ใช้ได้เร็วที่สุดพร้อมกันทั้งคู่ รถ B จะถูกเลือกก่อน C นอกจากนี้การจองจะให้ความสำคัญกับคำสั่งจองที่มาก่อนเสมอ สำหรับการจองแต่ละครั้ง ผู้จองจะได้รับคำตอบกลับมาว่าจะได้ใช้รถคันใด ซึ่งมีเกณฑ์การเลือกรถเป็นไปตามที่อธิบายไว้ก่อนหน้านี้

จงเขียนโปรแกรมที่รับจำนวนคำสั่งจอง N และคำสั่งจองทั้ง N คำสั่ง จากนั้นคำนวณว่ารถคันใดจะถูกนำไปใช้กับคำสั่งจองแต่ละคำสั่งโดยหากเป็นรถ A โปรแกรมจะพิมพ์ข้อความว่า A และขึ้นบรรทัดใหม่ ถ้าเป็นรถ B หรือ C ก็พิมพ์ผลลัพธ์ออกมาในลักษณะเดียวกัน กำหนดเพิ่มเติมว่าในตอนแรกรถตู้ทั้งสามคันว่างและพร้อมใช้ทั้งหมด

ข้อมูลเข้า

1. บรรทัดแรกระบุจำนวนคำสั่งจองเป็นจำนวนเต็มบวก N โดยที่ $1 \leq N \leq 10,000$
2. บรรทัดที่ 2 ถึง $N+1$ ระบุคำสั่งจองเรียงตามลำดับการขอ (บรรทัดที่มาก่อนหมายถึงจองก่อน) ในแต่ละบรรทัดประกอบด้วยเลขจำนวนเต็มบวกหนึ่งตัวคือ t โดยที่ $1 \leq t \leq 15$ (นั่นคือจองรถตู้ได้ครั้งละ 1 ถึง 15 วัน)

ผลลัพธ์

มีทั้งหมด N บรรทัด โดยแต่ละบรรทัดระบุว่ารถคันใดจะถูกนำไปใช้กับคำสั่งจองแต่ละคำสั่ง โดยผลลัพธ์เรียงตามลำดับคำสั่งจอง

ตัวอย่าง

ตัวอย่างที่หนึ่ง		ตัวอย่างที่สอง		ตัวอย่างที่สาม	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
6	A	6	A	7	A
3	B	1	B	2	B
1	C	2	C	2	C
2	B	2	A	1	C
2	C	1	A	1	A
2	A	1	B	1	B
1		3		3	C
				1	

อธิบายตัวอย่างที่หนึ่ง

บรรทัดแรก: เลข 6 ในคือจำนวนคำสั่งจองที่จะต้องคำนวณ

บรรทัดที่สอง: เลข 3 คือจำนวนวันที่จะใช้รถ เนื่องจากในตอนแรกรถทุกคันว่างหมดจึงเลือกใช้รถ A ตรงนี้ควรจำไว้ด้วยว่ารถ A จะว่างใช้อีกทีในวันที่ 4

บรรทัดที่สาม: เลข 1 คือจำนวนวันที่จะใช้รถ เนื่องจากตอนนี้รถที่ว่างใช้ได้เร็วที่สุดคือ B และ C จึงเลือก B ก่อน เช่นเดิมจำไว้ด้วยว่า B จะว่างใช้อีกทีในวันที่ 2

บรรทัดที่สี่: เลข 2 คือจำนวนวันที่จะใช้รถ เนื่องจากตอนนี้รถที่ว่างใช้ได้เร็วที่สุดคือ C จึงเลือก C เช่นเดิมจำไว้ด้วยว่า C จะว่างใช้อีกทีในวันที่ 3

บรรทัดที่ห้า: เลข 2 คือจำนวนวันที่จะใช้รถ เนื่องจากตอนนี้รถที่ว่างใช้ได้เร็วที่สุดคือ B จึงเลือก B เช่นเดิมจำไว้ด้วยว่า B จะว่างใช้อีกทีในวันที่ $2 + 2 = 4$

บรรทัดที่หก: เลข 2 คือจำนวนวันที่จะใช้รถ เนื่องจากตอนนี้รถที่ว่างใช้ได้เร็วที่สุดคือ C จึงเลือก C เช่นเดิมจำไว้ด้วยว่า C จะว่างใช้อีกทีในวันที่ $3 + 2 = 5$

บรรทัดที่เจ็ด: เลข 1 คือจำนวนวันที่จะใช้รถ เนื่องจากตอนนี้รถที่ว่างใช้ได้เร็วที่สุดคือ A และ B จึงเลือก A

ปัญหา 4 คัดเลือกนักร้อง [the_voice_su]

ในการประกวดร้องเพลงรายการ the_voice_su มีกรรมการอยู่สองคน คนแรกเป็นชายและคนที่สองเป็นหญิง กรรมการแต่ละคนจะทำการคัดเลือกผู้ที่เข้ารอบ โดยกระบวนการคัดเลือกเป็นดังนี้

1. ผู้เข้าประกวดจะมาร้องเพลงที่ละคนเรียงจากหมายเลข 1 จนถึงหมายเลข N
2. กรรมการทั้งสองท่านจะให้คะแนนเป็นจำนวนเต็มระหว่าง 0 ถึง 10
3. กรรมการจะพยายามคัดเลือกผู้ประกวดให้เข้ามาอยู่ในความดูแลของตนเพื่อฝึกฝนและเข้าสู่รอบต่อไป นอกจากนี้ กรรมการแต่ละคนจะดูแลผู้เข้าประกวดได้ไม่เกิน K คน ผู้ที่เข้ารอบในการประกวดคือผู้ที่ได้รับการดูแลจากกรรมการ
4. ในการให้คะแนน ถ้ากรรมการท่านใดให้คะแนน 9 หรือ 10 กับผู้เข้าประกวด จะหมายความว่าผู้เข้าประกวดคนดังกล่าวมีความสามารถในระดับสูง และกรรมการท่านนั้นต้องการรับมาดูแลด้วย อย่างไรก็ตามกรรมการท่านนั้นจะต้องมีนักร้องในความดูแลของตนยังไม่ครบ K คน ไม่เช่นนั้นก็จะรับผู้เข้าประกวดมาดูแลไม่ได้
5. ในบางครั้งกรรมการทั้งสองคนต่างเห็นว่าผู้เข้าประกวดมีความสามารถในระดับสูงและต้องการรับผู้ประกวดคนเดียวกันมาอยู่ในความดูแล ถ้ากรรมการทั้งสองยังมีคนไม่ครบ K คน ผู้เข้าประกวดจะเลือกกรรมการที่เป็นชายหรือหญิงเช่นเดียวกันตน แต่หากมีกรรมการเพียงหนึ่งท่านที่ยังรับผู้เข้าประกวดมาดูแลได้ ผู้เข้าประกวดก็จะอยู่ในความดูแลของกรรมการคนดังกล่าวโดยอัตโนมัติ ในกรณีที่เลวร้ายที่สุดก็คือกรรมการทั้งสองท่านไม่สามารถรับผู้เข้าประกวดมาดูแลเพิ่มได้ ผู้เข้าประกวดก็จะถูกรอ ไม่ว่าจะได้คะแนนดีสักเพียงใดก็ตาม
6. การคัดเลือกจะเรียงตามหมายเลขผู้เข้าประกวดตามที่ได้รับไว้ ทำให้ผู้ที่มีหมายเลขอันดับท้าย ๆ มีสิทธิ์น้อยลงที่จะได้รับการคัดเลือก เพราะกรรมการทั้งสองท่านอาจจะได้ผู้เข้าประกวดในความดูแลครบแล้วก็ได้ อย่างไรก็ตามทางผู้จัดการแข่งขันเห็นว่า อย่างน้อยควรให้ผู้เข้าประกวดทุกคนได้แสดงความสามารถให้เห็นแก่สาธารณชน ผู้เข้าประกวดทุกคนจึงจะร้องเพลงไปจนครบทุกคนและกรรมการก็จะยังทำการให้คะแนนกับผู้เข้าแข่งขันทุกคนตามปกติ

จงเขียนโปรแกรมที่ทำการคำนวณว่านักร้องหมายเลขใดบ้างที่เข้ารอบและผู้ที่ยังรออยู่ในความดูแลของกรรมการท่านใด

ข้อมูลเข้า

1. บรรทัดแรกเป็นจำนวนเต็มบวก N และ K ตามลำดับโดยที่ $1 < N \leq 10,000$ และ $1 \leq K \leq 1,000$
2. บรรทัดที่ 2 ถึง N + 1 ระบุเพศและคะแนนของผู้เข้าประกวด หนึ่งในผู้เข้าประกวดต่อหนึ่งบรรทัด เรียงจากผู้เข้าประกวดหมายเลขที่หนึ่งไปจนหมายเลขที่ N แต่ละบรรทัดมีตัวเลขจำนวนเต็มสามตัวซึ่งระบุเพศและข้อมูลคะแนนของผู้เข้าประกวด ตัวเลขแรกเป็นเพศ ซึ่ง 1 แทนเพศชายและ 2 แทนเพศหญิง ตัวเลขที่สองและสามในบรรทัดแทนคะแนนจากกรรมการคนที่หนึ่งและสองตามลำดับ

ผลลัพธ์

มีทั้งหมดไม่เกิน 2K บรรทัด โดยแต่ละบรรทัดระบุหมายเลขของผู้เข้าประกวดที่เข้ารอบ ตามด้วยกรรมการที่รับผู้เข้าประกวดไว้ในความดูแล โดยใช้เลข 1 แทนกรรมการคนแรกซึ่งเป็นชาย และเลข 2 แทนกรรมการคนที่สองซึ่งเป็นหญิง การเรียงลำดับการแสดงผลนั้น ให้เรียงตามหมายเลขของผู้เข้าประกวด **หมายเหตุ** เป็นไปได้ว่าจำนวนผู้

เข้าประกวดที่เข้ารอบอาจจะมีน้อยกว่า 2K ถ้ากรรมการเห็นว่าผู้เข้าประกวดที่มีความสามารถในระดับสูงมีไม่มากนัก
ตัวอย่าง

ตัวอย่างที่หนึ่ง		ตัวอย่างที่สอง		ตัวอย่างที่สาม	
ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
6 2	1 1	7 2	2 2	7 3	1 2
1 9 10	2 2	1 8 7	4 1	1 8 9	2 2
1 8 10	3 2	2 9 9	5 2	2 9 9	5 2
2 9 9	5 1	1 7 8	6 1	1 8 7	6 1
2 8 9		1 10 8		1 7 8	
2 10 9		2 10 9		2 9 10	
1 10 10		2 9 9		1 10 10	
		1 9 9		2 8 9	

อธิบายตัวอย่างที่หนึ่ง

บรรทัดแรก: เลข 6 คือจำนวนผู้เข้าประกวด เลข 2 คือค่า K ซึ่งแสดงจำนวนผู้เข้าประกวดที่กรรมการสามารถรับมาดูแลเพื่อฝึกฝนและให้เข้ารอบต่อไปได้

บรรทัดที่สอง: เลข 1 แสดงว่าผู้เข้าประกวดเป็นชาย ส่วน 9 และ 10 คือคะแนนจากกรรมการคนที่หนึ่งและสองตามลำดับ จากคะแนนแสดงว่ากรรมการทั้งสองท่านต้องการรับผู้เข้าประกวดคนนี้ได้ เนื่องจากมีที่ว่างพร้อมกัน ผู้เข้าประกวดจึงเลือกที่จะอยู่กับกรรมการคนแรกเพราะเป็นผู้ชายเหมือนกัน

บรรทัดที่สาม: เลข 1 แสดงว่าผู้เข้าประกวดเป็นชาย ส่วน 8 และ 10 คือคะแนนจากกรรมการคนที่หนึ่งและสองตามลำดับ จากคะแนนแสดงว่ามีเฉพาะกรรมการคนที่สองที่ต้องการรับผู้เข้าประกวดคนนี้ได้ เนื่องจากกรรมการคนที่สองยังมีที่ว่าง ผู้เข้าประกวดหมายเลขสองจึงเข้ารอบและอยู่กับกรรมการคนที่สอง

บรรทัดที่สี่: เลข 2 แสดงว่าผู้เข้าประกวดเป็นหญิง ส่วน 9 และ 9 คือคะแนนจากกรรมการคนที่หนึ่งและสองตามลำดับ จากคะแนนแสดงว่ากรรมการทั้งสองท่านต้องการรับผู้เข้าประกวดคนนี้ได้ และมีที่ว่างพร้อมกัน ผู้เข้าประกวดจึงเลือกที่จะอยู่กับกรรมการคนที่สองเพราะเป็นผู้หญิงเหมือนกัน

บรรทัดที่ห้า: เลข 2 แสดงว่าผู้เข้าประกวดเป็นหญิง ส่วน 8 และ 9 คือคะแนนจากกรรมการคนที่หนึ่งและสองตามลำดับ จากคะแนนแสดงว่ามีเฉพาะกรรมการคนที่สองที่ต้องการรับผู้เข้าประกวดคนนี้ได้ แต่กรรมการที่สองยังไม่มีที่ว่างอีกแล้ว ผู้เข้าประกวดจึงตรอไปตามกติกา

บรรทัดที่หก: จากคะแนนแสดงว่ากรรมการทั้งสองท่านต้องการรับผู้เข้าประกวดคนนี้ได้ แต่มีเฉพาะกรรมการคนแรกที่ยังมีที่ว่าง ผู้เข้าประกวดจึงอยู่ในความดูแลของกรรมการคนที่หนึ่ง

บรรทัดที่เจ็ด: จากคะแนนแสดงว่ากรรมการทั้งสองท่านต้องการรับผู้เข้าประกวดคนนี้ได้ แต่ไม่มีใครมีที่ว่าง ผู้เข้าประกวดจึงตรอไปตามกติกาไม่ว่าจะได้คะแนนมากสักเพียงใดก็ตาม

หมายเหตุ โปรแกรมของเราไม่จำเป็นที่จะต้องรับอินพุตครบทุกบรรทัดก็ได้

ปัญหา 5 จัดตารางสูตรคูณ [mult_table]

จงเขียนโปรแกรมที่พิมพ์ตารางสูตรคูณจากตัวเลขที่ผู้ใช้ใส่เข้าไป โดยผู้ใช้จะใส่ตัวเลขจำนวนเต็มบวกที่มีค่าตั้งแต่ 1 ถึง 99 เข้าไป จากนั้นโปรแกรมก็จะพิมพ์ตารางสูตรคูณจากตัวเลขที่ผู้ใช้ใส่เข้าไป โดยคูณเลขหนึ่งถึงสิบสองพร้อมจัดช่องว่างให้สวยงาม แต่ถ้าผู้ใช้ใส่เลขที่อยู่นอกขอบเขต 1 ถึง 99 โปรแกรมจะพิมพ์ว่า “Number is out of range” (กำหนดให้การคูณต้องทำด้วยลูป ห้ามใช้วิธีการ copy-paste code) **หมายเหตุ** การจัดช่องว่างต่าง ๆ ของผลลัพธ์ในข้อนี้ต้องเป็นไปอย่างเป็นระบบ ขอให้อ่านคำแนะนำทางด้านท้ายตัวอย่างเพื่อทำความเข้าใจเพิ่มเติม

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
5	5 x 1 = 5 5 x 2 = 10 5 x 3 = 15 5 x 4 = 20 5 x 5 = 25 5 x 6 = 30 5 x 7 = 35 5 x 8 = 40 5 x 9 = 45 5 x 10 = 50 5 x 11 = 55 5 x 12 = 60	25	25 x 1 = 25 25 x 2 = 50 25 x 3 = 75 25 x 4 = 100 25 x 5 = 125 25 x 6 = 150 25 x 7 = 175 25 x 8 = 200 25 x 9 = 225 25 x 10 = 250 25 x 11 = 275 25 x 12 = 300	99	99 x 1 = 99 99 x 2 = 198 99 x 3 = 297 99 x 4 = 396 99 x 5 = 495 99 x 6 = 594 99 x 7 = 693 99 x 8 = 792 99 x 9 = 891 99 x 10 = 990 99 x 11 = 1089 99 x 12 = 1188

คำแนะนำ

- สังเกตให้ดีว่าตัวเลขถูกจัดเรียงขีดขวาที่หลักหน่วยของผลลัพธ์
- วิธีที่จะทำแบบนี้ได้มีมากกว่าหนึ่งวิธี แต่วิธีที่แนะนำคือการใช้ตัวเลขหน้า %d เช่น %3d จะทำให้โปรแกรมพิมพ์เลขออกมาในลักษณะที่ว่า ‘ถ้าเลขมีน้อยกว่าสามหลัก จะปะช่องว่างไว้ที่ด้านหน้าของตัวเลข’ และนี่คือที่มาของช่องว่างหน้า 25, 50, และ 75 ในตัวอย่างที่ข้อมูลเข้าคือ 25
- ตัวอย่างการใช้ตัวเลขหน้า %d (อันนี้เป็นแบบที่ใช้กับเลข 25) คือ

```
printf("%d x %2d = %3d\n", x, mul, x*mul);
```
- แต่ถ้าเราใช้ %2d, %3d, หรือ %4d อยู่ตลอด มันจะทำให้มีช่องว่างมากเกินไปหรือน้อยเกินไปได้ แนะนำว่าให้ใช้ if-else มาตั้งเงื่อนไขว่าจะใช้เลขกี่หลัก และควรเลือกตามจำนวนหลักของผลคูณที่มีค่ามากที่สุด

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่แปด

การซ้อนลูป

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

จุดประสงค์การเรียนรู้: เพื่อสร้างความคุ้นเคยกับแนวคิดของการใช้ลูปสองชั้น

ปัญหา 1 กระหน้าพิมพ์ดอกจัน [AsteriskBurst]

จงใช้ลูปสองชั้นรับเลขจำนวนเต็มบวก R และ C โดยที่ R และ C คือจำนวนแถวและหลัก (คอลัมน์) ตามลำดับ จากนั้นโปรแกรมจะพิมพ์พื้นที่สี่เหลี่ยมที่มีดอกจัน R แถวและ C คอลัมน์ออกมา

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
3 4	**** **** ****
4 3	*** *** *** ***

ปัญหา 2 นับเลขแบบต่อแถว [RowCounting]

จงใช้ลูปสองชั้นรับเลขจำนวนเต็มบวก R และ C โดยที่ R และ C คือจำนวนแถวและหลัก (คอลัมน์) ตามลำดับ จากนั้นโปรแกรมจะนับเลขจาก 1 จนถึง $R \times C$ เช่นถ้า $R = 3$ และ $C = 4$ โปรแกรมจะนับเลขจาก 1 ถึง 12 นอกจากนี้ตัวเลขจะถูกพิมพ์ออกมาเป็นแถว แถวละ C ค่า จากในตัวอย่างเดิม ก็จะได้ผลลัพธ์เป็น

1 2 3 4

5 6 7 8

9 10 11 12

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
3 4	1 2 3 4 5 6 7 8 9 10 11 12
4 3	1 2 3 4 5 6 7 8 9 10 11 12

ปัญหา 3 พิมพ์พิกัดของแถวและคอลัมน์ [RCCoord]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็มบวก R และ C โดยที่ R และ C แทนหมายเลขแถวและหมายเลขหลัก (คอลัมน์) สูงสุดที่เราต้องการแสดงผล สำหรับปัญหานี้ ให้นักศึกษาใช้ลูปสองชั้นเพื่อเขียนพิกัดจุดที่เป็นเลขจำนวนเต็มทั้งหมด จากตำแหน่ง (0, 0) ไปจนถึงตำแหน่ง (R, C) โดยรวมตำแหน่ง (R, C) นี้ด้วย ในการพิมพ์พิกัดนั้น แต่ละบรรทัดจะมีหมายเลขแถวเท่ากันและมีส่วนพิกัดของหลักจะไล่จากน้อยไปมากโดยเริ่มจากเลขศูนย์ ดังแสดงข้างล่าง

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
2 3	(0, 0) (0, 1) (0, 2) (0, 3) (1, 0) (1, 1) (1, 2) (1, 3) (2, 0) (2, 1) (2, 2) (2, 3)
3 2	(0, 0) (0, 1) (0, 2) (1, 0) (1, 1) (1, 2) (2, 0) (2, 1) (2, 2) (3, 0) (3, 1) (3, 2)

หมายเหตุ ข้อมูลเข้าจะเป็นเลขจำนวนเต็มบวกแน่ ๆ ดังนั้นไม่ต้องคอยตรวจว่า R และ C เป็นจำนวนเต็มบวก

ปัญหา 4 กราฟแท่งแนวนอน [HorizontalBarGraph]

จงเขียนโปรแกรมที่รับเลขจำนวนเต็มเข้ามาจนกว่าจะพบเลขศูนย์หรือติดลบ โปรแกรมนี้จะพิมพ์เครื่องหมายดอกจันออกมาเป็นจำนวนตามตัวเลขที่ผู้ใช้ใส่เข้ามา (ยกเว้นตัวเลขศูนย์หรือติดลบที่ใช้ระบุจุดสิ้นสุดข้อมูล) โดยโปรแกรมจะพิมพ์เครื่องหมายดอกจันออกมาหนึ่งชุดต่อตัวเลขหนึ่งตัว เมื่อได้ตัวเลขใหม่ก็จะพิมพ์เครื่องหมายดอกจันในบรรทัดถัดมา เช่น ถ้าผู้ใช้ใส่เลข 5 เข้ามาในบรรทัดแรก และเลข 3 ในบรรทัดที่สอง โปรแกรมก็จะพิมพ์เครื่องหมายดอกจันมา 5 ตัวในบรรทัดแรกและ 3 ตัวในบรรทัดที่สอง ผลลัพธ์ที่ได้จะเหมือนกับกราฟแท่งในแนวนอนดังแสดงในตัวอย่าง

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
5 7 3 0	***** ***** ***
9 2 10 4 -2	***** ** ***** ****

ปัญหา 5 ระบายพื้นที่สี่เหลี่ยมขนมเปียกปูน [Rhombus]

จงเขียนโปรแกรมที่รับค่าตัวเลขจำนวนเต็มคือบวก N จากนั้นโปรแกรมจะระบายสี่เหลี่ยมขนมเปียกปูนด้วยเครื่องหมายดอกจัน * โดยที่สี่เหลี่ยมที่โปรแกรมจะระบายนี้มีความยาวเส้นผ่านศูนย์กลางเท่ากับ N ดังตัวอย่างข้างล่าง

ตัวอย่าง

ข้อมูลเข้า (N)	ผลลัพธ์
5	* *** ***** *** *
7	* *** ***** ***** ***** *** *

หมายเหตุ ข้อมูลเข้าจะเป็นเลขคี่บวกแน่ ๆ ดังนั้นโปรแกรมไม่ต้องคอยตรวจว่า N เป็นเลขคี่

******* ข้อนี้นี้เป็นแบบฝึกหัดที่ต้องใช้หัวคิดและ/หรือทักษะในการสังเกตรูปแบบของผลลัพธ์

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่เก้า

อาเรย์หนึ่งมิติ

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

คำสั่ง

1. ให้เขียนโปรแกรมภาษาซีสำหรับปัญหาที่ให้ไป
2. ระเบียบการส่งงานเขียนโปรแกรมก็คือนักศึกษาจะส่งโค้ดเข้าไปโปรแกรมตรวจงาน

จุดประสงค์การเรียนรู้ การใช้อาเรย์หนึ่งมิติคู่กับการใช้ลูปทั้งแบบชั้นเดียวและสองชั้น

ปัญหา 1 พิมพ์เลขย้อนกลับ [ReversePrint]

จงเขียนโปรแกรมที่รับค่า N มาค่าหนึ่ง จากนั้นโปรแกรมจะรับเลขจำนวนเต็มมาอีก N ค่า เมื่อรับเสร็จแล้วโปรแกรมจะพิมพ์เลขทั้งหมดออกมาย้อนหลังไปหน้า โดยเลขแต่ละตัวเว้นด้วยช่องว่างหนึ่งช่อง

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
5 1 2 3 4 5	5 4 3 2 1
6 -2 5 7 0 -1 4	4 -1 0 7 5 -2

หมายเหตุ ข้อมูลเข้าตัวแรกจะเป็นจำนวนเต็มบวกแน่ ๆ ดังนั้นไม่ต้องคอยตรวจว่า N เป็นจำนวนเต็มบวกมีค่าไม่เกิน 10,000

ปัญหา 2 การแข่งขันประเภททีม [TeamCompetition]

ในการแข่งขันเกมประเภททีมเกมหนึ่ง ผู้เล่นจะถูกแบ่งออกเป็นสองทีม ทีมละ N คน คนที่ 1 ของทีมแรกจะแข่งกับคนที่ 1 ของทีมที่สอง ในทำนองเดียวกัน คนที่ 2 ของแต่ละทีมก็จะมาแข่งกัน และเป็นเช่นนี้จนครบทั้งหมด N คู่ กติกามีอยู่ว่าผู้เข้าแข่งขันแต่ละคนจะทำแต้มตั้งแต่ 0 ถึง 10 คะแนน ถ้าใครทำคะแนนมากกว่าคู่แข่งของตัวเองจะถือว่าชนะในเกมนั้น ถ้าทำได้น้อยกว่าถือว่าแพ้ และถ้าทำได้เท่ากันถือว่าเสมอ

การชนะในแต่ละเกมจะทำคะแนนให้กับทีมของผู้ชนะ 2 คะแนน การเสมอได้ 1 คะแนน และการแพ้จะไม่ได้คะแนน ทีมที่มีคะแนนรวมมากกว่าเป็นฝ่ายชนะ ทีมที่มีคะแนนรวมน้อยกว่าแพ้ และ หากทั้งสองทีมได้คะแนนรวมเท่ากันถือว่าเป็นผลเสมอ

จงเขียนโปรแกรมที่รับแต้มของผู้เข้าแข่งขันแต่ละคนและสรุปผลการแข่งขันประเภททีมออกมาให้ได้ตามข้อกำหนดเรื่องข้อมูลเข้าและผลลัพธ์ต่อไปนี้

ข้อมูลเข้า

1. บรรทัดแรกเป็นตัวเลขจำนวนเต็ม N ซึ่งระบุจำนวนผู้เข้าแข่งขันของแต่ละทีม ($N \leq 1000$)
2. บรรทัดที่สองคือแต้มที่ผู้เข้าแข่งขันในทีมที่หนึ่งทำได้ เรียงจากคนที่ 1 ไปคนที่ 2, 3, ..., N โดยแต้มเป็นจำนวนเต็ม แต้มของแต่ละคนถูกคั่นด้วยช่องว่าง
3. บรรทัดที่สามคือแต้มที่ผู้เข้าแข่งขันทีมที่สองทำได้ เรียงจากคนแรกไปคนสุดท้ายในลักษณะเดียวกับคะแนนของทีมที่หนึ่ง

ผลลัพธ์

1. บรรทัดแรกระบุทีมที่ชนะ [ดูรูปแบบผลลัพธ์ในตัวอย่างทางด้านใต้]
2. บรรทัดที่สองระบุคะแนนของทั้งสองทีม โดยนำคะแนนของทีมที่ชนะขึ้นก่อน ในกรณีที่เสมอให้แสดงคะแนนของทีมไหนก่อนก็ได้ (เพราะคะแนนเท่ากันลำดับจึงไม่มีผล)

ตัวอย่าง

ตัวอย่าง	ข้อมูลเข้า	ผลลัพธ์
1	5 1 2 3 1 4 3 9 3 5 6	Team 2 wins Score 9 to 1 [คำอธิบาย: ทีมที่สอง ชนะ 4 ครั้ง เสมอ 1 ครั้ง จึงมี 9 แต้ม ส่วนทีมแรก เสมอครั้งเดียวนอกนั้นแพ้หมดจึงมีเพียง 1 แต้ม]
2	5 7 9 3 5 6 3 8 7 8 5	Team 1 wins Score 6 to 4 [คำอธิบาย: ทีมที่หนึ่ง ชนะ 3 ครั้ง แพ้ 2 ครั้ง จึงมี 6 แต้ม ส่วนทีมที่สอง ชนะ 2 ครั้ง แพ้ 3 ครั้ง จึงมี 4 แต้ม]
3	7 7 8 7 8 5 5 9 9 9 7 4 4 6 6	Draw game Score 7 to 7 [คำอธิบาย: ทั้งสองทีมชนะ 3 แพ้ 3 เสมอ 1 มี 7 แต้มเท่ากัน ผลการแข่งขันจึงออกมาเป็นเสมอ]

คำแนะนำ ข้อนี้จะใช้อาเรย์หนึ่งมิติหนึ่งหรือสองอันก็ได้ สำหรับหลายคนอาจจะรู้สึกว่าการใช้อาเรย์หนึ่งมิติสองอันดูเป็นธรรมชาติมากกว่า ในที่นี้จะอธิบายวิธีคิดที่มีอาเรย์สองอันคือ A และ B

1. อ่านค่า N เข้ามา
2. วนรับค่าแถวแรกเข้าไปเก็บไว้ใน A
3. วนรับค่าแถวที่สองเข้าไปเก็บไว้ใน B (คือตอนนี้จะมีรูปหนึ่งชั้นสองอันแยกกัน)
4. วนลูปอีกครั้งอ่านค่าจาก $A[i]$ และ $B[i]$ และนำค่าทั้งสองมาเปรียบเทียบกัน
5. นำผลการเปรียบเทียบที่ได้ไปคิดเป็นคะแนนของแต่ละทีม

ปัญหา 3 ตำแหน่งของเลขที่สนใจ [NumberOccurrence]

กำหนดชุดตัวเลข $x_1, x_2, x_3, \dots, x_N$ เช่น 7, 9, 2, 9, 7, 10, 2, 9, 3, 9 ในปัญหานี้ผู้ใช้สนใจที่จะหาว่า ตัวเลขที่ตนสนใจปรากฏอยู่ลำดับที่เท่าใดบ้าง เช่น ถ้าสนใจเลข 9 จากตัวอย่างเราก็จะได้ว่าเลข 9 ปรากฏเป็นลำดับที่ 2, 4, 8 และ 10 ส่วนถ้าสนใจเลข 7 ลำดับการปรากฏของมันก็จะเป็ลำดับที่ 1 และ 5

จงเขียนโปรแกรมที่รับชุดตัวเลข เมื่อรับชุดตัวเลขเสร็จแล้ว ให้รับตัวเลขที่ผู้ใช้สนใจ จากนั้นโปรแกรมจะต้องพิมพ์ลำดับการปรากฏของตัวเลขที่ผู้ใช้สนใจออกมาทั้งหมด ในกรณีที่ตัวเลขที่ผู้ใช้สนใจไม่ปรากฏอยู่ในชุดตัวเลขเลย ให้พิมพ์เลข 0 ออกมา

ข้อมูลเข้า

- บรรทัดแรกคือจำนวนเต็ม N ซึ่งเป็นตัวระบุความยาวของชุดตัวเลข โดยที่ $1 < N \leq 2,500,000$ (ความยาวคือจำนวนค่าตัวเลข เช่น ในตัวอย่างข้างบนจะมีค่า N เป็น 10)
- บรรทัดที่สองเป็นเลขจำนวนเต็มในชุดตัวเลขทั้งหมด โดยที่ $0 < x_i < 1,000$ เลขแต่ละค่าถูกคั่นด้วยช่องว่างหนึ่งช่อง
- บรรทัดที่สามเป็นเลขจำนวนเต็มหนึ่งตัวที่ผู้ใช้สนใจจะตรวจสอบดูว่ามันปรากฏอยู่ที่ลำดับเท่าใดบ้าง

ผลลัพธ์

มีหนึ่งบรรทัดแสดงลำดับการปรากฏของตัวเลขที่สนใจ ค่าลำดับแต่ละค่าจะถูกคั่นด้วยช่องว่างหนึ่งช่อง แต่ถ้าเลขที่ผู้ใช้สนใจไม่ปรากฏในลำดับเลย โปรแกรมจะพิมพ์เลขศูนย์ออกมาเพียงตัวเดียว

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
10 7 9 2 9 7 10 2 9 3 9 7	1 5
10 7 9 2 9 7 10 2 9 3 9 9	2 4 8 10
15 3 5 7 9 2 3 4 7 3 2 1 100 333 777 100 100	12 15
15 3 5 7 9 2 3 4 7 3 2 1 100 333 777 100 6	0

คำแนะนำ เนื่องจากอาเรย์มีขนาดใหญ่ เราควรสร้างมันไว้ภายนอกฟังก์ชัน main ไม่เช่นนั้นพื้นที่หน่วยความจำภายใน main จะไม่เพียงพอที่จะสร้างอาเรย์ได้

ปัญหา 4 บันไดตัวเลข (NumberStairs2)

ในปัญหานี้ เราต้องการพิมพ์ข้อความออกมาเป็นรูปบันได โดยบันไดแต่ละชั้นเป็นตัวเลขจำนวนเต็มหนึ่งหลัก เรียงต่อกันไป [ดูตัวอย่างท้ายโจทย์ประกอบ] จำนวนชั้นถูกกำหนดด้วยจำนวนเลขโดดที่รับเข้ามา โดยโปรแกรมของเราจะต้องรับเลขโดด¹ ที่มีค่าตั้งแต่ 0 ถึง 9 เข้ามาเรื่อย ๆ จนกว่าจะพบเลขติดลบหรือเลขที่มีค่าตั้งแต่ 10 ขึ้นไป จากนั้นจึงพิมพ์บันไดตัวเลขออกมาเป็นผลลัพธ์จากเลขโดดที่ผู้ใช้ใส่เข้ามา

ข้อมูลเข้า

มีบรรทัดเดียว เป็นจำนวนเต็มไม่ทราบจำนวนที่แน่นอนล่วงหน้า ทราบแต่เพียงว่ามีไม่เกิน 501 ตัวและมีไม่น้อยกว่า 2 ตัว โดยตัวสุดท้ายจะเป็นเลขติดลบหรือไม่ก็เลขที่มีค่าตั้งแต่ 10 ขึ้นไป ส่วนเลขก่อนถึงตัวสุดท้ายเป็นเลขโดดมีค่าตั้งแต่ 0 ถึง 9 [รวมเลข 0 และ 9 ด้วย] เลขแต่ละตัวคั่นด้วยช่องว่างหนึ่งช่อง

ผลลัพธ์

เป็นบันไดตัวเลขที่พิมพ์ออกมาเป็นข้อความหลายบรรทัด โดยบรรทัดชั้นบนสุดมีตัวเลขเพียงตัวเดียว ส่วนชั้นต่ำลงมาจะมีจำนวนตัวเลขเพิ่มขึ้นเรื่อย ๆ ชั้นละหนึ่งตัว มีลักษณะดังแสดงในตัวอย่างด้านใต้ **หมายเหตุ** การพิมพ์ช่องว่างเป็นสิ่งสำคัญต้องมีจำนวนที่ถูกต้องพอดี และในผลลัพธ์ที่ถูกต้องนั้น บรรทัดสุดท้ายจะไม่มีช่องว่างหน้าตัวเลขเลย

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
3 5 1 0 -2	3 55 111 0000
0 1 5 3 10	0 11 555 3333
7 8 2 7 3 0 2 7 0 523	7 88 222 7777 33333 000000 2222222 77777777 000000000

คำแนะนำ ข้อนี้ใช้อาเรย์หนึ่งมิติก็จริง แต่ต้องใช้ลูบสองชั้นในการคำนวณ

1 เลขโดดคือจำนวนที่มีเลขอยู่แค่ตัวเดียว

ปัญหา 5 พิมพ์กราฟแท่งแนวตั้ง เวอร์ชันพื้นฐาน (vertical_bar_graph_basic)

การวาดกราฟเป็นการแสดงความมากน้อยของจำนวนที่มีความสำคัญมาก แม้ในปัจจุบันการแสดงผลแบบกราฟฟิกจะเป็นแนวทางหลักในการวาดกราฟ แต่แนวคิดของการวาดหลายประการก็ยังมีลักษณะเดียวกันกับการวาดด้วยตัวอักษร

จงเขียนโปรแกรมที่พิมพ์กราฟแท่งจำนวน N แท่ง โดยความสูงของกราฟแต่ละแท่งจะเป็นไปตามความมากน้อยของค่าที่ต้องการแสดง เช่น หากค่าที่ต้องการแสดงคือ 12 กราฟแท่งของเลข 12 นี้ก็จะมีจำนวนเครื่องหมายดอกจันทั้งหมด 12 อัน แต่ถ้าหากค่าที่ต้องการแสดงคือ 8 ก็จะมีเครื่องหมายดอกจันของกราฟแท่งดังกล่าวเพียง 8 อัน เป็นต้น

ข้อมูลเข้า

- บรรทัดแรกคือจำนวนเต็ม N โดยที่ $N \leq 100$
- บรรทัดที่สองคือจำนวนเต็มบวก N ตัว แต่ละตัวแทนความสูงของกราฟแต่ละแท่ง เรียงจากซ้ายไปขวา เลขแต่ละตัวคั่นด้วยช่องว่างหนึ่งช่อง นอกจากนี้เลขแต่ละตัวมีค่าไม่เกิน 200

ผลลัพธ์

เป็นกราฟแท่งที่ความสูงของแต่ละแท่งถูกแทนด้วยจำนวนเครื่องหมายดอกจันในแท่งนั้น ๆ นอกจากนี้กราฟแท่งสองอันที่อยู่ติดจากกันจะไม่มีช่องว่างมาคั่น (ดูตัวอย่างประกอบ)

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
5 4 5 8 1 3	* * * ** *** *** * *** * *****
6 1 3 2 4 3 3	* * *** ***** *****

หมายเหตุ ตัวอย่างชี้ให้เห็นว่ากราฟทุกแท่งจะมีฐานอยู่บรรทัดเดียวกันทั้งหมด และบรรทัดแรกของผลลัพธ์ที่พิมพ์ออกมาจะต้องมีเครื่องหมายดอกจันของกราฟที่สูงที่สุดด้วย

คำแนะนำ ข้อนี้ควรใช้อาเรย์หนึ่งมิติ ส่วนลูปนั้นจะมีอยู่สองชุด คือชุดที่ใช้อ่านข้อมูลเข้าจะเป็นลูปชั้นเดียว แต่ตอนพิมพ์กราฟจะเป็นลูปอีกชุดซึ่งเป็นลูปสองชั้น

ปัญหา 6 หาที่ตั้งร้านขายของริมทาง (road_side_shop)

นักลงทุนรายหนึ่งต้องการเปิดร้านขายของบนถนนที่มีความยาวทั้งหมด N ช่วง แต่ในช่วงจะมีประชากรอยู่เป็นจำนวนต่าง ๆ กันไป เขามีสมมติฐานว่า โดยปรกติแล้วลูกค้าที่จะมาซื้อของที่ร้านจะอยู่ไม่ไกลจากร้านเกินระยะ K ช่วงถนน และเพื่อให้ร้านมีลูกค้าได้มากที่สุด เขาจึงพยายามหาตำแหน่งของร้านที่จะครอบคลุมช่วงถนนให้ได้จำนวนประชากรรวมมากที่สุดเท่าที่เป็นได้ และสิ่งแรกที่เขาต้องการทราบให้ได้ก่อนก็คือว่า จำนวนลูกค้าที่มากที่สุดที่จะมาซื้อของมีกี่คน

จงเขียนโปรแกรมที่คำนวณหาจำนวนลูกค้าที่มากที่สุดที่จะมาซื้อของที่ร้าน โดยจำนวนลูกค้านับจากจำนวนประชากรบนช่วงถนนที่อยู่ห่างจากร้านไม่เกิน K ช่วงถนน [มีตัวอย่างและคำอธิบายด้านล่างประกอบความเข้าใจ]

ข้อมูลเข้า

- บรรทัดแรกคือจำนวนเต็ม N และ K โดยที่ $1 \leq N \leq 10,000$ และ $1 \leq K \leq 100$
- บรรทัดที่สองเป็นจำนวนเต็มบวกหรือศูนย์แสดงจำนวนประชากรในถนนแต่ละช่วง เรียงจากช่วงแรกไปช่วงสุดท้ายตามลำดับ เลขแต่ละตัวคั่นด้วยช่องว่างหนึ่งช่อง และมีค่าไม่เกิน 500

ผลลัพธ์

เป็นเลขจำนวนเต็มแสดงจำนวนประชากรรวมสูงสุดที่เป็นไปได้ ซึ่งอยู่ในบริเวณที่จะมาซื้อของที่ร้าน

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
5 1 3 2 1 0 3	6
8 1 0 4 4 1 0 5 0 5	10
8 2 0 4 4 1 0 5 0 5	14

อธิบายตัวอย่างที่หนึ่ง ถนนมีอยู่ทั้งหมด 5 ช่วง ขอบเขตการเดินทางมาใช้บริการที่ร้านของลูกค้าจะไม่เกิน 1 ช่วงถนน ดังนั้นถ้าตั้งร้านไว้ที่ช่วงถนนที่สอง ประชากรที่อยู่ในช่วงที่หนึ่ง สอง และสามจะมาซื้อของได้ ดังนั้นจำนวนประชากรรวมในกรณีที่ตั้งร้านไว้ ณ ช่วงถนนที่สองจึงเป็น $3 + 2 + 1 = 6$ แต่ถ้าตั้งร้านในช่วงถนนที่สามจะได้จำนวนลูกค้าเป็น $2 + 1 + 0 = 3$ ถ้าตั้งในช่วงถนนที่สี่จะได้จำนวนลูกค้าเป็น $1 + 0 + 3 = 4$ ดังนั้นการตั้งร้านในช่วงถนนที่สองจะได้จำนวนลูกค้ารวมสูงสุดคือ 6 คน ผลลัพธ์จึงเป็น 6 ทั้งนี้ขอให้สังเกตด้วยว่าการตั้งร้านที่ติดขอบหรือใกล้ช่วงถนนด้านปลายทั้งสองมากเกินไปจะไม่มีทางให้ผลลัพธ์ที่ดีกว่าเดิม เราจึงไม่จำเป็นต้องนำผลลัพธ์เหล่านั้นมาพิจารณาด้วย

คำแนะนำ ควรใช้อาเรย์หนึ่งมิติและลูปสองชั้นในการแก้ปัญหา และให้ระมัดระวังที่ชี้ช่องอาเรย์มีค่าติดลบหรือเกินกว่าขอบเขตของอาเรย์ ความผิดพลาดของค่าดัชนีจะทำให้โปรแกรมค้างและเกิดเป็น time out ได้

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สิบ

อาเรย์สองมิติ

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

คำสั่ง

1. ให้เขียนโปรแกรมภาษาซีสำหรับปัญหาที่ให้ไป
2. ระเบียบการส่งงานเขียนโปรแกรมก็คือนักศึกษาจะส่งโค้ดเข้าไปโปรแกรมตรวจงาน

จุดประสงค์การเรียนรู้ การใช้อาเรย์สองมิติร่วมกับการใช้ลูปสองชั้น

ปัญหา 1 การทรานสโพสเมตริกซ์ [transpose _matrix]

การทรานสโพสเมตริกซ์เป็นการดำเนินการทางเมตริกซ์ที่มีการใช้งานอย่างกว้างขวาง การทรานสโพสคือการสลับข้อมูลในเมตริกซ์จากแถวที่ r คอลัมน์ที่ c ไปแถวที่ c คอลัมน์ที่ r การสลับตำแหน่งนี้ หากเมตริกซ์ไม่ใช่เมตริกซ์จัตุรัส จะทำให้ขนาดของเมตริกซ์ผลลัพธ์เปลี่ยนไปด้วย

จงเขียนโปรแกรมที่คำนวณผลลัพธ์การทรานสโพสเมตริกซ์ A โดยที่เมตริกซ์ A นี้มีจำนวนแถวไม่เกิน 5 และมีจำนวนหลักไม่เกิน 5 เช่นกัน กำหนดให้เมตริกซ์นี้เก็บผลลัพธ์เป็นเลขจำนวนเต็มเท่านั้น

ข้อมูลเข้า

1. เลขจำนวนเต็มบวกสองตัวแรกคือจำนวนแถว R และจำนวนคอลัมน์ C ตามลำดับ
2. เลขจำนวนเต็ม C จำนวนที่แทนข้อมูลแถวแรกใน A จากซ้ายไปขวา
3. เลขจำนวนเต็มของแถวที่เหลือใน A ซึ่งรับเข้ามาทีละแถว แถวละ C จำนวนในลักษณะเดียวกันกับข้อ b

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
3 3 1 2 3 4 5 6 7 8 9	1 4 7 2 5 8 3 6 9
2 4 1 2 3 4 5 6 7 8	1 5 2 5 3 7 4 8

คำแนะนำ

1. ควรใช้อาเรย์สองมิติมาเก็บข้อมูล เรารู้ขนาดมากที่สุดที่เป็นไปได้ของข้อมูล เราจึงควรเตรียมอาเรย์ที่รองรับขนาดสูงสุดที่เป็นไปได้นั่นไว้
2. การรับข้อมูลเข้าควรใช้ลูปสองชั้น (ดูตัวอย่างหน้า 30-31 ในซีทเรื่องอาเรย์เป็นแนวทาง)

3. ผลลัพธ์ไม่จำเป็นต้องนำไปใส่ไว้ในอาเรย์ใหม่ เราเปลี่ยนวิธีวนลูปและลำดับการเข้าถึงข้อมูลก็เพียงพอแล้ว (จะสร้างอาเรย์ผลลัพธ์ขึ้นมาก็ได้ แต่ขั้นตอนมันจะยุ่งยากซับซ้อนยิ่งกว่า)

ปัญหา 2 ผลรวมความแตกต่างของเมตริกซ์ [sum_matrix_diff]

จงเขียนโปรแกรมที่รับข้อมูลเมตริกซ์จัตุรัสขนาด $N \times N$ จากผู้ใช้ จากนั้นให้นับผลรวมความแตกต่างของคู่ตรงข้ามระหว่างข้อมูลในซีกขวาบน กับข้อมูลในซีกซ้ายล่าง เช่น จากเมตริกซ์ขนาด 4×4

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 13 & 14 & 15 & 16 \\ 9 & 10 & 11 & 12 \\ 5 & 6 & 7 & 8 \end{bmatrix}$$

การจับคู่ตรงข้ามเพื่อหาผลรวมความแตกต่างได้ผลเป็นดังนี้

$$|2 - 13| + |3 - 9| + |15 - 10| + |4 - 5| + |16 - 6| + |12 - 7| = 11 + 6 + 5 + 1 + 10 + 5 = 38$$

ข้อมูลเข้า

- บรรทัดแรกระบุจำนวนเต็ม N (จำนวนแถวและคอลัมน์ของเมตริกซ์) โดยที่ $N \leq 100$
- บรรทัดที่ 2 ระบุข้อมูลแถวแรกของเมตริกซ์ (ข้อมูลในเมตริกซ์เป็นเลขจำนวนเต็มทั้งหมด) ข้อมูลแต่ละตัวคั่นด้วยช่องว่าง
- บรรทัดที่ 3 ถึง $N + 1$ ระบุข้อมูลในเมตริกซ์แถวที่ 2, 3, ..., N ตามลำดับ

ผลลัพธ์

ตัวเลขแสดงผลรวมความแตกต่างของตัวเลขคู่ตรงข้ามทั้งหมดในเมตริกซ์

ตัวอย่าง

ตัวอย่าง	ข้อมูลเข้า	ผลลัพธ์
1	3 1 2 3 3 9 8 7 3 7	10
2	4 1 2 3 4 13 14 15 16 9 10 11 12 5 6 7 8	38
3	5 1 2 3 1 4 3 9 3 5 6 7 8 7 8 5 9 9 7 4 4 7 4 2 3 0	32

ปัญหา 3 นับยอดเขา [mountain_top]

จงเขียนโปรแกรมที่รับความสูงของพื้นที่ โดยพื้นที่นี้ถูกแบ่งออกเป็นพื้นที่ย่อยจำนวน $N \times N$ ช่อง (N แถว คูณ N คอลัมน์) และข้อมูลความสูงที่รับมาจะเป็นความสูงของพื้นที่ย่อยในแต่ละช่อง จากข้อมูลความสูงนี้ เราต้องการหายอดเขาภายในพื้นที่ย่อย ซึ่งยอดเขาอยู่ในพื้นที่ย่อยที่มีคุณสมบัติดังนี้

1. พื้นที่ย่อยไม่อยู่ตรงบริเวณขอบของพื้นที่ทั้งหมดในข้อมูล
2. พื้นที่ย่อยมีความสูงที่มากที่สุดภายในอาณาบริเวณขนาด 3×3 ช่อง โดยพื้นที่ย่อยที่มียอดเขาเป็นศูนย์กลางของอาณาบริเวณขนาด 3×3 ช่องนี้
3. ภายในอาณาบริเวณขนาด 3×3 ในข้อสอง ไม่มีพื้นที่ย่อยใดที่มีความสูงเทียบเท่ากับมัน (ยอดเขาต้องอยู่ในพื้นที่ย่อยที่สูงที่สุดโดยไม่มีพื้นที่ย่อยอื่นสูงเท่ากับมันภายในอาณาบริเวณขนาด 3×3 ดังกล่าว)

โปรแกรมจะต้องทำการนับยอดเขาทั้งหมดที่พบ แสดงความสูงและตำแหน่งของแต่ละยอดเขา

ข้อมูลเข้า

1. บรรทัดแรกระบุจำนวนเต็ม N (จำนวนช่องพื้นที่ตามแนวตั้งและแนวนอน) โดยที่ $N \leq 1,000$
2. บรรทัดที่ 2 ระบุข้อมูลความสูงของพื้นที่ย่อยแถวแรก เรียงจากซ้ายไปขวา ความสูงแต่ละค่าถูกคั่นด้วยช่องว่าง (ข้อมูลความสูงทั้งหมดเป็นจำนวนเต็มบวกหรือศูนย์)
3. บรรทัดที่ 3 ถึง $N + 1$ ระบุความสูงของพื้นที่ย่อยแถวที่สองถึงแถวที่ N ตามลำดับ

ผลลัพธ์

1. บรรทัดแรกระบุจำนวนยอดเขาทั้งหมดที่อยู่ในพื้นที่
2. บรรทัดที่ 2 ระบุข้อมูลของยอดเขาเรียงตามลำดับดังนี้ ความสูงของยอดเขา หมายเลขแถว หมายเลขคอลัมน์ (หมายเลขแถวและคอลัมน์เริ่มนับจากศูนย์)
3. หากมียอดเขามากกว่าหนึ่ง ยอดเขาที่เหลือจะถูกแสดงข้อมูลออกมาในบรรทัดต่อมา หนึ่งบรรทัดต่อหนึ่งยอดเขา

หมายเหตุ ข้อมูลเข้ารับประกันว่าจะต้องมียอดเขาอย่างน้อยหนึ่งยอดในพื้นที่

ตัวอย่าง

ตัวอย่างที่	ข้อมูลเข้า	ผลลัพธ์
1	4 1 5 7 9 1 8 7 6 1 9 2 5 5 7 8 2	1 9 2 1
2	5 1 2 3 1 4 3 9 3 5 6 7 8 7 8 5 9 9 7 4 4 7 4 2 3 0	2 9 1 1 8 2 3
3	7 1 5 7 5 4 7 4 1 8 7 6 8 0 3 1 9 2 5 7 6 5 5 9 8 2 5 8 1 2 6 4 9 4 2 2 5 9 6 4 7 5 4 4 5 3 2 4 5 6	4 8 1 4 8 3 5 9 4 3 9 5 1

ปัญหา 4 พิมพ์แผนที่ลานจอดรถ [car_park_map]

ลานจอดรถแห่งนี้มีขนาด M แถว N คอลัมน์ ลานจอดรถนี้ใช้ระบบอัตโนมัติในการตรวจหาว่าตำแหน่งใดที่มีรถจอดอยู่ อย่างไรก็ตามสิ่งที่ระบบระบุออกมาจะเป็นตำแหน่งพิกัดแถวและคอลัมน์ที่เครื่องตรวจพบว่ามีรถจอด ซึ่งทำให้เข้าใจได้ยากกว่าตำแหน่งใดบ้างที่เป็นที่ว่างหรือมีรถจอด

เจ้าของที่จอดรถจึงได้ขอให้โปรแกรมเมอร์ทำการวาดแผนที่ลานจอดรถจากข้อมูลพิกัดที่เครื่องตรวจที่จอดรถส่งมาให้ โดยแผนที่นี้จะแสดงเครื่องหมายขีดเส้นใต้ `_` เพื่อระบุว่าตำแหน่งดังกล่าวไม่มีรถจอด และแสดงเครื่องหมายกากบาท `x` (ใช้ตัวเอ็กซ์เล็ก) เพื่อแสดงว่ามีรถจอดอยู่ เช่น หากแผนที่คือ

```
x _ x x
  _ x _
  x x _
```

แสดงว่าลานจอดรถมีขนาด 3 แถว 4 คอลัมน์ โดยแถวแรกมีรถจอดที่คอลัมน์ที่หนึ่ง ที่สาม และ ที่สี่ ส่วนแถวที่สองมีรถจอดที่คอลัมน์ที่สามเพียงคันเดียว และแถวสุดท้ายมีรถจอดอยู่สองคันที่คอลัมน์ที่สามและสี่ จงเขียนโปรแกรมที่รับขนาดลานจอดรถและตำแหน่งที่มีรถจอด เพื่อทำการแปลงให้เป็นแผนที่ในลักษณะเดียวกับที่แสดงไว้ข้างบน

ข้อมูลเข้า

- บรรทัดแรกเป็นจำนวนเต็มบวก M และ N ตามลำดับ คั่นด้วยช่องว่าง โดยที่ $1 \leq M, N \leq 100$
- บรรทัดที่สองคือจำนวนเต็มบวก K ระบุจำนวนตำแหน่งพิกัดที่มีรถจอดอยู่ โดยที่ $K \leq M \cdot N$
- บรรทัดที่สามถึง $K + 2$ เป็นคู่ลำดับ $R C$ แสดงตำแหน่งแถวและคอลัมน์ที่มีรถจอดอยู่ หนึ่งแถวต่อหนึ่งคู่ลำดับ โดยที่ $1 \leq R \leq M$ และ $1 \leq C \leq N$

หมายเหตุ ตำแหน่งที่มีรถจอดอยู่อาจจะซ้ำกันได้ ไม่ว่าจะซ้ำกี่ครั้งก็ถือว่าไม่มีรถจอดอยู่ตำแหน่งนั้น

ผลลัพธ์

เป็นแผนที่แสดงการจอดรถ โดย `x` แทนตำแหน่งที่มีรถจอด และ `_` แสดงตำแหน่งที่ว่าง ในลักษณะเดียวกับที่กล่าวไว้ข้างต้น

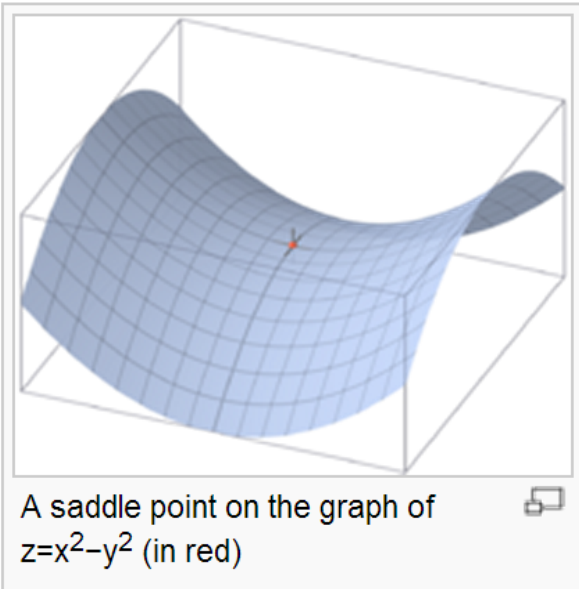
ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
3 4	x _xx	5 7	_____x
6	____x_	7	_____x_
1 1	____xx_	4 3	xx_____
1 3		3 2	____x_____
1 4		4 3	____x_____
2 3		1 7	
3 2		3 1	
3 3		5 2	
		2 6	

คำแนะนำ ข้อนี้ควรใช้อาเรย์สองมิติขนาด M แถว N คอลัมน์ โดยเริ่มแรกให้ช่องข้อมูลในอาเรย์ทั้งหมดมีค่าเป็นศูนย์ จากนั้นเมื่อได้พิกัดตำแหน่งรถมา ก็ให้เปลี่ยนช่องข้อมูลในอาเรย์ให้กลายเป็นหนึ่ง ทั้งนี้ควรใส่ใจด้วยว่าตำแหน่งพิกัดที่เครื่องตรวจแจ้งมาเริ่มจากหนึ่งในขณะที่หมายเลขช่องอาเรย์เริ่มจากศูนย์

ปัญหา 5 จุดอานม้า [saddle_point]

จุดอานม้าเป็นจุดที่ค่าของฟังก์ชันมีค่าสูงสุดในทิศทางหนึ่งแต่กลับเป็นค่าต่ำสุดในอีกทิศทางหนึ่ง การหาจุดอานม้าเป็นสิ่งสำคัญในการแก้ไขปัญหาทงวิทยาศาสตร์และวิศวกรรมหลายอย่าง (ข้อนี้โค้ดยาวพอสมควร และอาจจะเป็นโจทย์แบบฝึกหัดในแล็บเพียงข้อเดียวในวิชานี้ที่ผมรู้สึกว่าต้องออกแรงกันบ้าง)



ชื่อของจุดนี้มาจากลักษณะทางเรขาคณิตที่จะอยู่ในบริเวณที่รูปร่างคล้ายอานม้าพอดี บางทีจุดนี้ก็ถูกเรียกว่าจุด minimax เพราะเป็นทั้งค่าต่ำสุดและสูงสุดในคราวเดียวกัน (ดูภาพประกอบทางซ้าย)

ในการคำนวณเรามักจะเก็บค่า z ซึ่งบอกระดับความสูงของกราฟไว้ในอาร์เรย์สองมิติ การทำอย่างนี้นำไปสู่การคำนวณที่เป็นระบบซึ่งเป็นสิ่งที่เราจะทำในแบบฝึกหัดนี้

ภาพจาก Wikipedia.org

จุดอานม้าหรือจุดมินิเมกซ์ในอาร์เรย์สองมิติคือสมาชิกในอาร์เรย์ที่มีคุณสมบัติข้อใดข้อหนึ่งต่อไปนี้

1. เป็นค่าสูงสุดในแถวแต่กลับเป็นค่าต่ำสุดในหลัก
2. เป็นค่าต่ำสุดในแถวแต่กลับเป็นค่าสูงสุดในหลัก

(ดัดแปลงมาจากโจทย์ใน Schaum's Outlines: ทฤษฎีและตัวอย่างโจทย์ การเขียนโปรแกรมด้วยภาษา C++)

จงเขียนโปรแกรมที่ทำการหาจุดอานม้าในอาร์เรย์ขนาด R แถว C คอลัมน์ จากนั้นให้พิมพ์ตำแหน่งจุดอานม้าพร้อมด้วยค่าที่จุดอานม้าออกมา ในกรณีที่อาร์เรย์มีจุดอานม้ามากกว่าหนึ่งจุดให้พิมพ์จุดอานม้าทั้งหมดออกมาในรูปแบบตามตัวอย่างข้างล่าง **หมายเหตุ** ข้อมูลเข้ามีลักษณะเดียวกับแบบฝึกหัดข้อที่แล้ว ในกรณีที่ไม่มีจุดอานม้าให้พิมพ์คำว่า None ออกมา

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
3 4 1 2 3 3 9 5 6 7 7 4 9 2	$(0, 2) = 3$ $(1, 1) = 5$
4 4 7 5 3 4 2 5 4 2 9 7 3 1 5 1 2 5	None

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สิบเอ็ด

ฟังก์ชันและตัวชี้

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

คำสั่ง

1. ให้เขียนโปรแกรมภาษาซีสำหรับปัญหาที่ให้ไป
2. ระเบียบการส่งงานเขียนโปรแกรมก็คือนักศึกษาจะส่งโค้ดเข้าไปโปรแกรมตรวจงาน

ปัญหา 1 ค่าเฉลี่ยและความแปรปรวน [average_and_variance1]

เขียนโปรแกรมเพื่อรับเลขจำนวนเต็ม 8 ตัวเก็บไว้ในอาร์เรย์ของฟังก์ชัน main จากนั้นให้ส่งอาร์เรย์ดังกล่าวไปคำนวณผลในฟังก์ชัน `double average(int* array)` เพื่อคำนวณค่าเฉลี่ยของข้อมูลในอาร์เรย์ จากนั้นให้ใช้ผลลัพธ์ที่ได้จากฟังก์ชัน `average` ไปเป็นพารามิเตอร์ของฟังก์ชัน

`double variance(int* array, double avg);`

เพื่อคำนวณค่าความแปรปรวนตามสูตรด้านล่าง ท้ายสุดให้พิมพ์ค่าเฉลี่ยและค่าความแปรปรวนออกมาในฟังก์ชัน main

หมายเหตุ ผลลัพธ์ที่ได้จากฟังก์ชัน `average` ถูกป้อนให้เป็นพารามิเตอร์ตัวที่สองของฟังก์ชัน `variance`

สมการสำหรับคำนวณความแปรปรวน

$$\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2$$
 โดยที่ N คือจำนวนข้อมูลซึ่งในที่นี้มีค่าเท่ากับ 8, x_i คือข้อมูลแต่ละตัวในอาร์เรย์ และ \bar{x} คือค่าเฉลี่ยของข้อมูลทั้ง 8 (หาได้มาจากฟังก์ชัน `average`)

ข้อมูลเข้า เป็นเลขจำนวนเต็มจำนวนแปดตัว จะเป็นเลขติดลบก็ได้

ผลลัพธ์ ค่าเฉลี่ยตามด้วยค่าความแปรปรวนของเลขจำนวนเต็มทั้งแปด เลขทั้งสองคั่นด้วยช่องว่าง ทศนิยมสองตำแหน่ง **หมายเหตุ** การคำนวณเลขทศนิยมในข้อนี้ให้ใช้ชนิดข้อมูลเป็นแบบ `double`

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
1 2 3 4 5 6 7 8	4.50 6.00
6 7 8 9 10 11 12 13	9.50 6.00
7 -2 5 3 -1 0 6 -4	1.75 16.50

ปัญหา 2 ค่าเฉลี่ยและความแปรปรวน [average_and_variance2]

เขียนโปรแกรมเพื่อรับเลขจำนวนเต็ม N ตัวเก็บไว้ในอาร์เรย์ของฟังก์ชัน `main` จากนั้นให้ส่งอาร์เรย์ดังกล่าวไปคำนวณผลในฟังก์ชัน `double average(int* array, int N)` เพื่อคำนวณค่าเฉลี่ยของข้อมูลในอาร์เรย์จากนั้นให้ใช้ผลลัพธ์ที่ได้จากฟังก์ชัน `average` ไปเป็นพารามิเตอร์ของฟังก์ชัน `double varaince(int* array, double avg, int N)`; เพื่อคำนวณค่าความแปรปรวน ทำยสุดให้พิมพ์ค่าเฉลี่ยและค่าความแปรปรวนออกมาในฟังก์ชัน `main`

หมายเหตุ ความแตกต่างจากข้อที่แล้วก็คือว่า ในข้อนี้ขนาดของอาร์เรย์ถูกกำหนดด้วยตัวแปร N ไม่ได้มีขนาดตายตัว

ข้อมูลเข้า บรรทัดแรกเป็นเลขจำนวนเต็ม N โดยที่ $1 < N \leq 100$ ระบุจำนวนข้อมูลที่จะใช้คำนวณค่าทางสถิติจำนวนแปดตัว ส่วนบรรทัดที่สองจะเป็นเลขจำนวนเต็ม N ตัวใช้สำหรับคำนวณค่าทางสถิติ เลขแต่ละตัวคั่นด้วยช่องว่าง

ผลลัพธ์ ค่าเฉลี่ยตามด้วยค่าความแปรปรวนของ เลขทั้งสองคั่นด้วยช่องว่าง มีทศนิยมสองตำแหน่ง

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
8 1 2 3 4 5 6 7 8	4.50 6.00
12 7 -2 5 3 -1 0 6 -4 1 9 5 -8	1.75 24.93

ปัญหา 3 ฟังก์ชันทดสอบว่ามีเลขที่ซ้ำกันอยู่ในอาเรย์หรือไม่ [isUnique]

จงเขียนฟังก์ชัน isUnique ซึ่งทำการตรวจสอบเลขในอาเรย์จำนวนเต็ม arData ว่าเลขทุกตัวไม่มีซ้ำกันเลยใช่หรือไม่ หากเลขทุกตัวในอาเรย์ไม่ซ้ำกันเลย อาเรย์จะคืนเลข 1 แต่หากมีเลขที่ซ้ำกันอยู่อย่างน้อยหนึ่งตัว โปรแกรมจะคืนเลข 0 ทั้งนี้ arData มีตัวเลขอยู่ทั้งหมด N ตัว สำหรับฟังก์ชัน isUnique มีแม่แบบดังนี้

```
int isUnique(int* arData, int N);
```

[มีตัวอย่างข้อมูลเข้าและผลลัพธ์อยู่ในหน้าถัดไป]

คำแนะนำ 1 ข้อนี้ควรมีการลูปสองชั้น ชั้นแรก (ลูปชั้นนอก) ทำการเลือกตัวที่เราต้องการตรวจสอบว่าซ้ำหรือไม่ ส่วนชั้นที่สอง (ลูปชั้นใน) ทำหน้าที่ตรวจสอบว่าตัวเลขที่เราสนใจนั้นซ้ำกับตัวอื่น ๆ ในอาเรย์หรือไม่

คำแนะนำ 2 ความยากสำหรับมือใหม่ก็คือตรงลูปด้านใน คือเราต้องทำการตรวจสอบโดยไม่เผลอไปเปรียบเทียบตัวเลขที่เราสนใจกับตัวของมันเอง (แล้วพลาดไปสรุปว่าเลขซ้ำกัน) วิธีป้องกันปัญหานี้มีสองแบบ

แบบแรกคือการป้องกันผ่านการกำหนดตำแหน่งเริ่มต้นและสิ้นสุดลูปแต่ละชั้น เพื่อรับประกันว่าการเปรียบเทียบจะไม่เกิดขึ้นกับอาเรย์ช่องเดียวกันเป็นอันขาด และจะไม่มีเปรียบเทียบที่ซ้ำซ้อนด้วย (วิธีนี้ดี เพราะรวดเร็ว และเป็นที่ยอมรับในหมู่ผู้มีประสบการณ์) สามารถดูตัวอย่างโค้ดได้จากด้านท้ายปัญหาข้อ 5

แบบที่สองคือการวนอาเรย์ช่องที่ k กับทุกช่องที่ไม่ใช่ k โดยป้องกันการเปรียบเทียบซ้ำกับ k ด้วยการ ใช้ if แบบง่าย ๆ ข้อดีของวิธีนี้คือมือใหม่รู้สึกเข้าใจง่ายว่าการเปรียบเทียบเกิดขึ้นโดยไม่มีเลขคู่ใดที่ถูกละเอียด แต่ข้อเสียก็คือมีการเปรียบเทียบเลขซ้ำซ้อนทำให้เสียเวลาในการคำนวณไปมากโดยไม่มีประโยชน์เชิงคำนวณเกิดขึ้นมา

เรื่องอื่น ๆ ถ้าคิดไม่ออกและจำเป็นต้องใช้ตัวช่วย ลองดูโครงสร้างของฟังก์ชันหลังข้อ 5 อย่างไรก็ตาม ผู้เรียนก็ไม่ควรหวังพึ่งตัวช่วย จนกว่าจะได้ลองคิดด้วยตนเองอย่างจริงจังแล้ว

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
8 1 2 3 4 5 6 7 8	1
12 7 2 5 3 7 0 6 4 1 9 5 8	0
12 6 4 1 9 5 8 7 2 5 3 7 0	0

หมายเหตุ จำนวนข้อมูลเข้าจะมีไม่เกิน 10,000 ค่า

ปัญหา 4 อนุพันธ์ในอาเรย์ [array_derivative]

การหาอนุพันธ์ในอาเรย์เป็นวิธีที่พบบ่อยในการประมวลผลภาพและสัญญาณไฟฟ้า โดยคำนวณได้จากการนำอาเรย์ช่องที่ i ลบด้วยช่องที่ $i - 1$ เช่น หากอาเรย์มีสามช่อง เราจะเอาช่องที่ 2 ลบด้วยช่องที่ 1 และช่องที่ 1 ลบด้วยช่องที่ 0 ส่วนช่องที่ 0 นั้น เราจะกำหนดให้ค่าเป็นศูนย์ (ค่าพิเศษเนื่องจากไม่มีช่องด้านซ้ายมาทำการลบ)

ยกตัวอย่าง หากอาเรย์นั้นมีข้อมูลเป็น 3, 5, และ 4 ดังภาพข้างล่างนี้

3	5	4
---	---	---

ผลการหาอนุพันธ์จะเก็บลงในอาเรย์ผลลัพธ์ (AOutput) ทำให้ข้อมูลภายใน Aoutput มีค่าเป็น 0, 2, และ -1 ตามลำดับ ซึ่งช่องข้อมูลหมายเลขหนึ่งได้ผลลัพธ์เป็นสอง มาจาก $5 - 3$ และช่องข้อมูลหมายเลขสองมาจาก $4 - 5 = -1$

จงเขียนโปรแกรมเพื่อรับเลขจำนวนเต็ม N ตัวเก็บไว้ในอาเรย์ โดยที่ $N \leq 100$ จากนั้นให้ส่งอาเรย์ดังกล่าวไปคำนวณผลในฟังก์ชัน

```
void derivative(int* AInput, int* AOutput, const int N);
```

ผลลัพธ์ที่ได้จะถูกเก็บไว้ในอาเรย์ AOutput และจะถูกนำกลับมาแสดงผลในฟังก์ชัน main

ข้อมูลเข้า

- บรรทัดแรกระบุจำนวนเต็ม N ที่ระบุจำนวนข้อมูลในอาเรย์
- บรรทัดที่สองระบุจำนวนเต็ม N ตัวที่เป็นข้อมูลในอาเรย์

ผลลัพธ์ เลขจำนวนเต็มในอาเรย์ผลลัพธ์ที่เก็บค่าอนุพันธ์ของอาเรย์ข้อมูลเข้า เลขแต่ละตัวในอาเรย์ถูกคั่นด้วยช่องว่างหนึ่งช่อง

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
6 1 7 3 5 2 4	0 6 -4 2 -3 2
8 7 8 3 2 -1 0 5 9	0 1 -5 -1 -3 1 5 4

ปัญหา 5 ฟังก์ชันนับการซ้ำของเลขที่มีค่าสูงสุดในอาร์เรย์ [count_max]

จงเขียนฟังก์ชัน countMax สำหรับหาว่าเลขที่มีค่ามากที่สุดในอาร์เรย์มีค่าซ้ำกันทั้งหมดกี่ค่า (ถ้ามีตัวเดียวถือว่าซ้ำกัน 1 ค่า) โดยฟังก์ชันนี้รับอาร์เรย์เลขจำนวนเต็ม arData ซึ่งมีความยาว N หลังจากนับจำนวนเลขค่ามากที่สุดที่ซ้ำกันแล้ว ฟังก์ชันจะคืนจำนวนครั้งที่ซ้ำกันของเลขค่ามากที่สุด ทั้งนี้ฟังก์ชันที่ต้องการมีแม่แบบดังนี้

```
int countMax(int* arData, int N);
```

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
10 1 1 1 7 7 3 5 5 2 4	2
8 7 8 3 2 -1 0 5 9	1
12 1 1 1 7 7 7 1 1 1 7 7 7	6

อธิบายตัวอย่างที่ 1 เลข 7 เป็นเลขที่มีค่ามากที่สุด และในข้อมูลชุดนี้มีเลข 7 อยู่สองครั้ง ผลลัพธ์จึงเป็นเลข 2

*** เราไม่ได้ต้องการนับเลขที่ค่าซ้ำกันมากที่สุด แต่เราต้องการนับการซ้ำของเลขที่มีค่ามากที่สุด

คำแนะนำ ควรมีลูปชั้นเดียวสองชุด ชุดแรกมีหน้าที่หาเลขที่มีค่าสูงสุดในอาร์เรย์ ส่วนชุดที่สองมีหน้าที่นำค่าสูงสุดที่หาได้ ไปตรวจดูว่ามีเลขที่ซ้ำกับมันกี่ครั้ง จำนวนครั้งที่นับได้คือผลลัพธ์ของโปรแกรมนี้

หมายเหตุ จำนวนข้อมูลเข้าจะมีไม่เกิน 10,000 ค่า

[มีตัวช่วยอยู่ในหน้าถัดไป]

ตัวช่วยข้อ 3: isUnique

```
int isUnique(_____ arData, int N) {  
    int p, q;  
    for(p = 0; p < _____; ++p) {  
        for(q = _____; q < N; ++q) {  
            if(arData[p] _____ arData[q]) {  
                return _____;  
            }  
        }  
    }  
    return _____;  
}
```

ตัวอย่างข้อ 5: countMax

```
int countMax(_____ arData, int N) {  
    int _____ = INT_MIN;  
    int i;  
    for(i = 0; i < N; ++i) {  
        if(arData[i] > _____) {  
            M = _____;  
        }  
    }  
    int _____ = 0;  
    for(i = 0; i < N; ++i) {  
        if(arData[i] _____) {  
            ++count;  
        }  
    }  
    return _____;  
}
```

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สิบสอง

เตรียมสอบอาเรย์

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

ปัญหา 1 พิมพ์เลขเดินหน้าหรือย้อนหลัง [forward_backward_print]

จงเขียนโปรแกรมรับค่าจำนวนเต็มจากผู้ใช้งาน N ค่า โดยที่ $N \leq 5,000$ และค่าแต่ละตัวที่ผู้ใช้ใส่เข้ามาจะเป็นเลขจำนวนเต็มบวก มีค่าตั้งแต่ 1 ถึง 10,000 (รวมเลข 1 และ 10,000 ด้วย) เมื่อผู้ใช้ใส่ค่าเข้ามาครบ N ค่าแล้ว โปรแกรมจะรับค่าจำนวนเต็ม M เพิ่มอีกหนึ่งค่า ถ้าค่า M ที่รับเพิ่มเข้ามาเป็นบวก โปรแกรมจะนำเลขที่ผู้ใช้ใส่เข้ามา N ตัวแรกบวก M แล้วพิมพ์เลขผลบวกเรียงจากตัวแรกไปตัวสุดท้าย แต่ถ้าเลขที่ผู้ใช้ใส่เข้ามาติดลบ โปรแกรมจะนำเลขที่ผู้ใช้ใส่เข้ามา N ตัวแรกบวก M เช่นเดิม แต่ลำดับการพิมพ์จะเรียงย้อนจากตัวสุดท้ายมาที่ตัวแรก

ข้อมูลเข้า

- บรรทัดแรกระบุค่า N โดยที่ $N \leq 5,000$
- บรรทัดที่สองระบุจำนวนเต็มบวกที่ผู้ใช้ใส่เพิ่มเข้ามาทั้งหมด N ค่า แต่ละค่าคั่นด้วยช่องว่างหนึ่งช่อง
- บรรทัดที่สามระบุค่า M เป็นจำนวนเต็มบวกหรือจำนวนเต็มลบ (จะไม่มีค่าศูนย์มาให้แน่นอน) โดยที่ $-10,000 \leq M \leq 10,000$

ผลลัพธ์

มีบรรทัดเดียว เป็นค่าตัวเลข N ตัวที่ผู้ใช้ใส่เข้ามาบวกกับค่า M เรียงจากตัวแรกไปตัวสุดท้ายหรือจากตัวสุดท้ายไปตัวแรก ขึ้นกับค่า M โดยเลขแต่ละตัวคั่นด้วยช่องว่างหนึ่งช่อง

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
5 7 8 9 5 4 1	8 9 10 6 5
5 7 8 9 5 4 -1	3 4 8 7 6
10 1 5 9 5 3 2 1 7 12 2 3	4 8 12 8 6 5 4 10 15 5
10 1 5 9 5 3 2 1 7 12 2 -4	-2 8 3 -3 -2 -1 1 5 1 -3

ปัญหา 2 นับคะแนนเลือกตั้ง [count_vote]

ในการเลือกตั้งครั้งหนึ่งมีผู้สมัครทั้งหมด N คน โดยผู้สมัครจะได้หมายเลข $1, 2, 3, \dots, N$ โดยที่ไม่ซ้ำกันเลย ในการเลือกตั้งครั้งนี้มีผู้ออกเสียงทั้งหมด K คน โดยผู้ออกเสียงแต่ละคนจะเลือกออกเสียงโหวตผู้สมัครหมายเลขใด หมายเลขหนึ่งเพียงคนเดียวเท่านั้น กำหนดให้การเลือกตั้งครั้งนี้ไม่มีผู้ใดที่งดออกเสียง และไม่มีบัตรเสียเลย กล่าวคือผู้ออกเสียงเลือกหมายเลขใดหมายเลขหนึ่งจาก 1 ถึง N อย่างถูกต้องทุกคน จงเขียนโปรแกรมที่ทำการนับคะแนนโหวต จากนั้นให้โปรแกรมรายงานว่าผู้ชนะคือหมายเลขใดและได้รับคะแนนโหวตเท่าใด

ข้อมูลเข้า

- บรรทัดแรกระบุจำนวนเต็ม N และ K ตามลำดับ คั่นด้วยช่องว่าง โดยที่ N มีค่าไม่เกิน 100 และ K ไม่น้อยกว่า 1
- บรรทัดที่สองประกอบด้วยตัวเลขจำนวนเต็ม K ตัว แต่ละตัวคือหมายเลขผู้สมัครที่ผู้ออกเสียงแต่ละคนเลือก เลขแต่ละตัวคั่นด้วยช่องว่างหนึ่งช่อง

ผลลัพธ์

- บรรทัดแรกระบุหมายเลขผู้สมัครที่ได้รับคะแนนโหวตมากที่สุด
- บรรทัดที่สองระบุคะแนนของผู้สมัครที่ได้รับคะแนนมากที่สุด

หมายเหตุ ข้อมูลเข้ารับประกันว่าผู้สมัครที่ได้รับคะแนนมากที่สุดจะมีเพียงคนเดียวเท่านั้น

คำแนะนำ

- ให้สร้างอาร์เรย์เก็บผลลัพธ์ขนาด 101 ช่องขึ้นมา สำหรับเก็บคะแนนโหวต (เราสร้างอาร์เรย์ขึ้นมาตามจำนวนช่องสูงสุดที่อาจจะได้ใช้ เพราะค่า N สูงสุดคือ 100)
- ให้ระวังว่าหมายเลขผู้สมัครเริ่มจากเลข 1 ไม่ใช่เลขศูนย์นั้นเป็นเหตุผลว่าทำไมจำนวนช่องของอาร์เรย์มันถึงเกินออกมาหนึ่งช่อง (อาจจะมีบางช่องที่ไม่ได้ใช้เลยก็ได้ ไม่ต้องประหลาดใจ)
- อาร์เรย์ที่สร้างขึ้นมาเก็บคะแนนโหวตที่ผู้สมัครแต่ละคนได้รับ ดังนั้นอย่าลืมว่าหลังจากสร้างอาร์เรย์ขึ้นมาแล้ว เราต้องทำให้คะแนนโหวตของผู้สมัครทุกคนเป็นศูนย์ก่อนจากนั้นจึงค่อยเริ่มนับคะแนน
- ในกรณีที่ผลลัพธ์จากโปรแกรมนักศึกษาผิด แนะนำให้ลองพิมพ์ค่าในอาร์เรย์จากช่องหมายเลข 1 ถึง N ออกมาดู
- พอนับคะแนนโหวตเสร็จแล้วให้วนดูว่าช่องไหนในอาร์เรย์ที่มีค่ามากที่สุด ช่องนั้นแหละที่นำไปสู่คำตอบของเรา

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
5 10 3 2 3 2 3 1 2 3 4 5	3 4
4 20 4 3 2 4 3 2 1 2 3 1 1 1 2 2 1 1 4 4 1 1	1 8
6 25 4 3 2 4 5 1 6 1 4 6 5 1 3 6 6 6 6 6 6 6 6 3 2 2 1	6 10

ปัญหา 3 แผนที่บอกลำดับเส้นทางอย่างง่าย [waypoint_order]

ในระบบแผนที่ปัจจุบัน เช่น Google Map ผู้ใช้จะทราบข้อมูลว่าต้องเดินทางไปจุดใดบ้างตามลำดับ โดยมีการแสดงออกมาให้เห็นบนภาพแผนที่ด้วย ในโจทย์ข้อนี้ นักศึกษาจะต้องทำการแสดงภาพแผนที่บอกลำดับอย่างง่ายตามเงื่อนไขของข้อมูลเข้าและผลลัพธ์ดังนี้

ข้อมูลเข้า

- บรรทัดแรกคือจำนวนเต็มบวกสองตัวระบุขนาดแผนที่ โดยตัวแรกบอกจำนวนแถว R และตัวที่สองระบุจำนวนคอลัมน์ C ทั้งนี้ $5 \leq R, C \leq 100$
- บรรทัดที่สองเป็นเลขจำนวนเต็มบวก $2 \leq K \leq 9$ ระบุจำนวนจุดที่ต้องเดินทางไปเพื่อถึงจุดหมาย
- บรรทัดที่ 3 ถึง $K + 2$ เป็นคู่ลำดับ (A, B) ของเลขจำนวนเต็ม หนึ่งคู่ต่อหนึ่งบรรทัด โดยที่ A ระบุตำแหน่งแถว และ B ระบุตำแหน่งคอลัมน์ของจุดที่ต้องเดินทางไป โดยที่ $1 \leq A \leq R$ และ $1 \leq B \leq C$ เช่น ถ้า $A = 3$ และ $B = 2$ หมายความว่า จะต้องเดินทางไปแถวที่ 3 และคอลัมน์ที่ 2

ทั้งนี้บรรทัดที่ 3 คือจุดแรกที่ต้องเดินทางไป บรรทัดที่ 4 เป็นจุดที่สอง บรรทัดที่ 5 เป็นจุดที่สามและเป็นเช่นนี้ไปเรื่อย ๆ จนครบ K จุด และในข้อมูลเข้า รับประกันว่าไม่มีคู่ลำดับที่ตำแหน่งซ้ำกันเลย

หมายเหตุ มีตัวอย่างและคำอธิบายตัวอย่างด้านท้ายของโจทย์

ผลลัพธ์

เป็นแผนที่สองมิติขนาด R แถวและ C คอลัมน์ ภายในจะมีเลขที่เป็นเป็นได้ทั้งหมด 10 คือ 0 ถึง 9 โดยเลข 1 คือจุดแรกที่ต้องเดินทางไป เลข 2 คือจุดที่สองที่ต้องเดินทางไป และเป็นเช่นนี้ไปเรื่อย ๆ จนครบ K จุด ส่วนเลข 0 คือตำแหน่งที่ไม่ได้เป็นจุดที่ต้องเดินทางไป (อาจจะเป็นระหว่างทางหรืออะไรก็ได้สุดแล้วแต่ ไม่ต้องใส่ใจประเด็นนั้น เพราะโจทย์ข้อนี้ต้องการพูดถึงแผนที่แบบง่าย) เลขแต่ละตัวในแต่ละแถวจะถูกคั่นด้วยช่องว่างหนึ่งช่อง

ตัวอย่างที่ 1

ข้อมูลเข้า	ผลลัพธ์
5 8	0 1 0 0 0 3 0 0
3	0 0 0 0 0 0 0 0
1 2	0 0 0 0 0 0 0 0
5 7	0 0 0 0 0 0 0 0
1 6	0 0 0 0 0 0 2 0

อธิบายตัวอย่างที่ 1 จากตัวอย่างนี้แผนที่จะมีทั้งหมด 5 แถวและ 8 คอลัมน์ มีจุดที่ต้องเดินทางไปสามจุด โดยจุดแรกอยู่ตำแหน่ง (1, 2) ซึ่งหมายถึงแถวที่ 1 คอลัมน์ที่ 2 [เลขแถวและคอลัมน์เริ่มนับจาก 1] ดังนั้นในแผนที่จึงพิมพ์เลข 1 ไว้ที่ตำแหน่งดังกล่าว ส่วนจุดที่สองที่ต้องเดินทางไปคือตำแหน่ง (5, 7) ตำแหน่งดังกล่าวจึงเป็นเลข 2 บนแผนที่ผลลัพธ์ที่พิมพ์ออกมา ในทำนองเดียวกัน จุดที่สามที่ต้องเดินทางไปคือ (1, 6) ตำแหน่งดังกล่าวจึงเป็นเลข 3 ส่วนตำแหน่งอื่น ๆ ให้พิมพ์ออกมาเป็นเลข 0

ตัวอย่างที่ 2

ข้อมูลเข้า	ผลลัพธ์
10 8	0 4 0 0 0 0 0 0
9	0 0 0 0 0 0 0 8
5 7	0 0 0 0 0 0 0 0
5 8	0 0 0 0 0 6 0 0
10 1	0 0 0 0 0 9 1 2
1 2	0 0 0 5 0 0 0 0
6 4	0 0 0 0 0 0 0 0
4 6	0 0 0 0 0 0 0 0
9 3	0 0 7 0 0 0 0 0
2 8	3 0 0 0 0 0 0 0
5 6	

คำแนะนำ ควรทำการคำนวณตามลำดับดังนี้

1. ให้สร้างอาเรย์สองมิติเก็บจำนวนเต็ม ซึ่งมีขนาดสูงสุดเป็น 100 แถวและ 100 คอลัมน์ขึ้นมา (จะทำเป็น 101 แทนก็ได้)
2. ใส่ค่า 0 เข้าไปในอาเรย์ดังกล่าวทั้งหมด
3. จากค่า A B แต่ละคู่ ให้ใส่เลขที่บอกลำดับที่ต้องเดินทางไปลงในตำแหน่งที่สอดคล้องกัน ทั้งนี้อย่าลืมว่าเลขแถวและคอลัมน์ในอาเรย์นับจาก 0 แต่เลขแถวและคอลัมน์ในแผนที่เริ่มจาก 1 (ยกเว้นคุณจะใช้วิธีสละแถวและคอลัมน์แรกทิ้งไป) *** ถ้าเราใช้ตัวนับ ตอนอ่านค่า A B แต่ละคู่ เราจะสามารถบอกลำดับที่ต้องใส่เข้าไปในอาเรย์ได้โดยง่ายแน่นอนว่าในครั้งนี้นี้ตัวนับควรเริ่มจากเลข 1
4. พิมพ์ค่าในอาเรย์สองมิติออกมาทั้งหมด โดยพิมพ์ออกมาทีละแถว

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่สิบสาม

สตรีค

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

คำสั่ง

1. ให้เขียนโปรแกรมภาษาซีสำหรับปัญหาที่ให้ไป
2. ระเบียบการส่งงานเขียนโปรแกรมก็คือนักศึกษาจะส่งโค้ดเข้าไปโปรแกรมตรวจงาน

ปัญหา 1 บันทึกและค้นข้อมูลนักศึกษา [data_record_and_retrieval]

ฝ่ายทะเบียนต้องการเก็บข้อมูลนักศึกษาสี่อย่างคือ

1. รหัสประจำตัวนักศึกษา เป็นข้อความยาว 8 ตัวอักขระพอดี
2. ชื่อ เป็นข้อความยาวไม่เกิน 30 ตัวอักขระ
3. นามสกุล เป็นข้อความยาวไม่เกิน 50 ตัวอักขระ
4. ชั้นปี เป็นเลขจำนวนเต็ม

ทั้งนี้กำหนดให้จำนวนนักศึกษามี N คน โดยที่ $N \leq 20,000$ คน ฝ่ายทะเบียนต้องการโปรแกรมที่บันทึกข้อมูลนักศึกษาพร้อมทั้งสามารถค้นคืนข้อมูลนักศึกษาโดยใช้ชั้นปีเป็นตัวกำหนดคือ เช่น หากฝ่ายทะเบียนต้องการค้นหานักศึกษาในชั้นปีหนึ่งทั้งหมด ฝ่ายทะเบียนจะใส่เลข 1 เข้าไป และโปรแกรมของเราจะต้องพิมพ์ข้อมูลนักศึกษาหนึ่งคนต่อหนึ่งบรรทัดโดยข้อมูลแต่ละคนจะเรียงจากรหัส ตามด้วยชื่อ และ นามสกุล ข้อมูลแต่ละอย่างคั่นด้วยช่องว่าง ลำดับการแสดงผลจะเรียงตามลำดับการบันทึกข้อมูลเข้า (ดูตัวอย่างด้านท้ายโจทย์ประกอบ)

ข้อมูลเข้า

1. บรรทัดแรกเป็นจำนวนเต็ม N ซึ่งแทนจำนวนนักศึกษาที่ต้องการบันทึกข้อมูล
2. บรรทัดที่สองถึง $N + 1$ เป็นข้อมูลนักศึกษาแต่ละคน หนึ่งคนหนึ่งบรรทัด โดยเรียงลำดับจาก รหัส ชื่อ นามสกุล และ ชั้นปี ตามลำดับ ทั้งนี้ข้อมูลแต่ละอย่างเว้นด้วยช่องว่าง และทั้งชื่อและนามสกุลต่างก็ไม่มีช่องว่าง (คือจะไม่มีชื่อนามสกุลจำพวก “ณ อยุธยา” หรืออะไรที่ทำให้มีช่องว่างอยู่ข้างในชื่อหรือนามสกุล)
3. บรรทัดที่ $N + 2$ คือจำนวนเต็ม Y ที่แทนเลขชั้นปีที่ต้องการค้นหา

ผลลัพธ์

เป็นรายการแสดงรหัส ชื่อ และ นามสกุล ของนักศึกษาทั้งหมดที่อยู่ในชั้นปีที่ Y ทั้งนี้หากไม่มีนักศึกษาในชั้นปีที่กำหนดให้พิมพ์คำว่า None ออกมาทางจอภาพ

ตัวอย่าง 1

ข้อมูลเข้า	ผลลัพธ์
3 07520101 Mhong Lim 4 07530102 Wanchana Munjai 3 07520103 Kalaya Tatong 4 4	07540101 Mhong Lim 07540103 Kalaya Tatong

ตัวอย่าง 2

ข้อมูลเข้า	ผลลัพธ์
3 07540101 Mhong Lim 3 07540102 Wanchana Munjai 4 07540103 Kalaya Tatong 3 1	None

คำแนะนำ

1. ควรมีการใช้สตรัคซึ่งมีสมาชิก 4 ตัวคือ (1) รหัส โดยมีชนิดข้อมูลเป็นสตริง, (2) ชื่อ เป็นสตริง, (3) นามสกุล เป็นสตริง และ (4) ชั้นปี เป็นจำนวนเต็ม ซึ่งเราสามารถที่จะเขียนการประกาศสตรัคนี้ได้เป็น

```
struct student {  
    char id[9];  
    char name[31];  
    char surname[61];  
    int year;  
} typedef Student;
```

2. ควรประกาศอาเรย์ของสตรัคนี้เพื่อให้เราสามารถเก็บข้อมูลต่าง ๆ ได้อย่างเป็นระบบและสามารถอ่านค่าซ้ำหลายรอบได้ นอกจากนี้สตรัคอาจมีขนาดใหญ่ การประกาศไว้แบบโกลบอลโดยระบุจำนวนไว้ให้เท่ากับจำนวนนักศึกษาสูงสุดที่จะยอมรับได้

```
Student A[20000];
```

*** คำว่าตัวแปรแบบโกลบอลคือตัวแปรซึ่งอยู่นอก main

ปัญหา 2 คะแนนเฉลี่ยของผู้เข้าสอบ [average_score]

[ข้อนี้เป็นพื้นฐานของโจทย์ข้อถัดไป ข้อนี้ใช้เพียงลูปหนึ่งชั้น ไม่จำเป็นต้องใช้อาเรย์ แต่ข้อต่อไปควรใช้อาเรย์]

ในการสอบคัดเลือกกรายการหนึ่ง ผู้ที่จะผ่านการสอบคัดเลือกจะต้องทำคะแนนได้ไม่ต่ำกว่าค่าเฉลี่ยของคะแนนผู้เข้าสอบทั้งหมด เพื่อที่จะหาว่าใครจะผ่านการสอบคัดเลือกบ้าง คณะกรรมการจึงเริ่มต้นจากการหาค่าเฉลี่ยก่อน ทั้งนี้ค่าเฉลี่ยจะคิดจากคะแนนที่เป็นเลขจำนวนเต็มมีค่าตั้งแต่ 0 ถึง 100 จากผู้เข้าสอบทั้ง N คน

จงเขียนโปรแกรมที่คำนวณค่าเฉลี่ยของคะแนนสอบ โดยคะแนนเฉลี่ยที่ได้จะอยู่ในรูปเลขทศนิยมแบบ double precision

ข้อมูลเข้า

- บรรทัดแรกคือจำนวนเต็ม N ซึ่งแทนจำนวนผู้เข้าสอบ โดยที่ $1 \leq N \leq 10,000$
- บรรทัดที่สองเป็นเลขจำนวนเต็มจำนวน N ตัวแทนคะแนนของผู้เข้าสอบแต่ละคน คะแนนแต่ละคะแนนคั่นด้วยช่องว่างหนึ่งช่องและมีค่าอยู่ในช่วง 0 ถึง 100 (รวมค่า 0 และ 100)

ผลลัพธ์

เป็นเลขทศนิยมแบบ double precision ที่เป็นค่าเฉลี่ยของคะแนนสอบ

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
3 10 7 8	8.333333
6 9 0 8 7 100 1	20.833333
7 8 8 9 9 7 10 10	8.714286
10 5 6 7 99 9 5 7 5 9 0	15.200000

คำแนะนำ การพิมพ์คำตอบให้ใช้ %f (เปอร์เซ็นต์ และ เอฟตัวเล็ก) การหาผลบวกจะทำด้วยเลขทศนิยมแบบ double precision ตั้งแต่แรกก็ได้

ปัญหา 3 จำนวน รหัส และ รายชื่อของผู้ที่เข้ารอบ [qualified_examinees]

ในการสอบคัดเลือกรายการหนึ่ง ผู้ที่จะผ่านการสอบคัดเลือกจะต้องทำคะแนนได้ไม่ต่ำกว่าค่าเฉลี่ยของคะแนนผู้เข้าสอบทั้งหมด เพื่อที่จะหาว่าใครจะผ่านการสอบคัดเลือกบ้าง คณะกรรมการจึงได้ทำการตรวจสอบว่าใครที่ได้คะแนนไม่น้อยกว่าค่าเฉลี่ยบ้าง จากนั้นจึงพิมพ์จำนวนและรหัสผู้ที่สอบผ่านทั้งหมดออกมาเป็นการประกาศผลสอบคัดเลือก

จงเขียนโปรแกรมที่นับจำนวนผู้สอบผ่านและรายงานรหัสประจำตัวผู้ที่สอบผ่านทั้งหมด

ข้อมูลเข้า

- บรรทัดแรกคือจำนวนเต็ม N ซึ่งแทนจำนวนผู้เข้าสอบ โดยที่ $1 \leq N \leq 10,000$
- บรรทัดที่ 2 ถึง $N + 1$ เป็นข้อมูลผู้เข้าสอบแต่ละคน หนึ่งคนต่อหนึ่งบรรทัด ซึ่งแต่ละบรรทัดประกอบด้วยรหัสประจำตัวและคะแนนสอบตามลำดับ คั่นด้วยช่องว่างหนึ่งช่อง โดยที่รหัสประจำตัวเป็นข้อความความยาว 5 ตัวอักษร และคะแนนสอบเป็นตัวเลขจำนวนเต็มอยู่ในช่วง 0 ถึง 100

ผลลัพธ์

- บรรทัดแรกเป็นเลขจำนวนเต็มระบุจำนวนผู้ที่สอบผ่านทั้งหมด (สมมติว่าผ่านทั้งหมด K คน)
- บรรทัดที่ 2 ถึง $K + 1$ เป็นรหัสประจำตัวผู้สอบผ่าน เรียงตามลำดับการปรากฏในข้อมูลเข้า

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
3 AB3A0 10 1B7Za 7 Bzab5 8	1 AB3A0
6 1B7Za 9 AB3A0 0 12345 8 67890 7 Bzab5 100 17A53 1	1 Bzab5
7 12345 8 67890 8 Bzab5 9 17A53 9 ZYA53 7 1B7Za 10 AB3A0 10	4 Bzab5 17A53 1B7Za AB3A0

(ถ้าคิดวิธีไม่ออก ลองอ่านคำแนะนำเกี่ยวกับการแก้ปัญหาในหน้าถัดไป แต่ถ้าคิดออก ก็ลงมือทำได้เลย)

คำแนะนำ เรื่องที่มีมือใหม่จะรู้สึกว่ายากมีสองประเด็นคือ

1. บางคนไม่รู้ว่าจะเก็บข้อมูลอย่างไรดี ซึ่งจุดนี้แก้ได้สองทางคือ ใช้สตรัคมาเก็บรหัสและคะแนนแต่ละคนไว้ หรือใช้อาเรย์สองชุด ชุดแรกเก็บเฉพาะรหัส ชุดที่สองเก็บเฉพาะคะแนน ทั้งสองทางนี้ถือว่าถูกต้องและใช้งานได้จริงทั้งคู่ แต่แบบที่สองจะสร้างความงุนงงสำหรับบางคน เนื่องจากอาเรย์จะมีลักษณะเป็นอาเรย์สองมิติ เพราะสตริงโดยตัวของมันเองก็เป็นอาเรย์หนึ่งมิติมาตั้งแต่แรก

2. บางคนคิดไม่ออกว่าจะทำอย่างไรถึงจะพิมพ์จำนวนผู้ที่สอบผ่านออกมา ก่อนแล้วค่อยพิมพ์รหัสของผู้สอบผ่านทั้งหมดออกมา ทางแก้ที่เป็นไปได้มีอยู่หลายทาง แต่ทางที่เขียนโปรแกรมเสร็จได้เร็วที่สุดก็คือการนับจำนวนก่อน แล้วก็พิมพ์จำนวนนั้นออกมา จากนั้นจึงค่อยทำการเริ่มตรวจอีกรอบถ้าใครคะแนนถึงก็พิมพ์รหัสของคนคนนั้นออกมาเลย นั่นคือรูปของการนับจำนวน กับรูปของการพิมพ์รหัสประจำตัวเป็นคนละรูปแยกจากกันต่างหาก

*** สิ่งแรกที่ต้องทำก็คือหาค่าเฉลี่ยของคะแนนสอบให้ได้ก่อน ซึ่งเราสามารถใช่วิธีจากข้อแรกมาดัดแปลงใช้ในข้อนี้ได้ แต่ก็ต้องเปลี่ยนวิธีเก็บข้อมูลด้วย เพราะข้อนี้ต้องใช้อาเรย์เนื่องจากการอ่านข้อมูลเข้าหลายรอบ

ปัญหา 4 หาผู้ได้คะแนนรวมสูงสุด [find_top_student]

วิชาแคลคูลัส 1 มีนักศึกษาอยู่ N คน และมีการสอบย่อยทั้งหมด k ครั้ง โดยที่ $N \leq 1000$ และ $k \leq 5$ อาจารย์ผู้สอนได้ทำการบันทึกคะแนนสอบของนักศึกษาทีละคน คือนำคะแนนสอบทั้งหมด k ครั้งของนักศึกษาค้นแรกบันทึกลงไปจนหมดก่อน แล้วจึงบันทึกคะแนนนักเรียนคนถัดมาทีละคนในลักษณะเดียวกัน ทั้งนี้อาจารย์ผู้สอนต้องการหาด้วยว่านักศึกษาที่ทำคะแนนได้สูงสุดนั้นได้คะแนนเท่าใด และมีการสอบย่อยกี่ครั้งที่นักศึกษาค้นดังกล่าวได้คะแนนสูงสุดในการสอบครั้งนั้น ๆ ด้วย ทั้งนี้สมมติให้ผู้ที่ทำคะแนนรวมสูงสุดมีเพียงคนเดียวเท่านั้น

ข้อมูลเข้า

- บรรทัดแรกเป็นเลขจำนวนเต็ม N และ k ตามลำดับ ข้อมูลทั้งสองคั่นด้วยช่องว่าง
- บรรทัดที่สองถึงบรรทัดที่ $N + 1$ เป็นคะแนนสอบของนักศึกษาแต่ละคน หนึ่งคนหนึ่งบรรทัด โดยคะแนนสอบเป็นเลขจำนวนเต็มเรียงจากครั้งที่หนึ่งถึงครั้งที่ k ตามลำดับ คั่นด้วยช่องว่าง

ผลลัพธ์

- บรรทัดแรกเป็นเลขจำนวนเต็มแสดงคะแนนรวมสูงสุด
- บรรทัดที่สองคือจำนวนการสอบที่นักศึกษาค้นที่ทำคะแนนรวมดีที่สุดในได้คะแนนสูงสุด

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
4 3	23	3 4	27
5 7 8	2	5 6 8 8	1
7 3 9		9 1 2 9	
8 6 9		1 7 8 4	
1 2 3			

หมายเหตุ ข้อนี้ไม่จำเป็นต้องใช้ struct การใช้ array สองมิติ นับเป็นทางออกที่เหมาะสมและเพียงพอ แต่ถ้าใครถนัดคิดแบบ struct คือสร้าง struct สำหรับเก็บคะแนนของนักเรียนคนหนึ่ง ๆ ขึ้นมา แล้วใช้ array หนึ่งมิติมาเก็บ struct ดังกล่าวก็ได้ ซึ่งวิธีนี้นับเป็นทางออกที่เหมาะสมเช่นกัน