

บทที่ 11 การจำลอง เหตุการณ์เพื่อแก้ปัญหาด้วย คอมพิวเตอร์

วัตถุประสงค์การเรียนรู้เชิงพฤติกรรม

- ✓ สามารถอธิบายความหมายของการจำลองเหตุการณ์ได้
- ✓ สามารถอธิบายบทบาทและความสำคัญของการจำลองเหตุการณ์ได้
- ✓ สามารถบอกประโยชน์ของการจำลองเหตุการณ์ได้
- ✓ สามารถบอกการประยุกต์ใช้การจำลองเหตุการณ์ ในปัจจุบันได้

สาเหตุที่ต้องใช้คอมพิวเตอร์จำลองเหตุการณ์

ยิ่งคอมพิวเตอร์มีความสามารถคิดคำนวณได้เร็วขึ้นเท่าไร คอมพิวเตอร์ก็จะยังสามารถช่วยให้มนุษย์สำรวจปัญหาต่าง ๆ ที่เราแทบจะไม่เคยทำได้มาก่อนได้ดียิ่งขึ้น หนึ่งในประโยชน์สำคัญของคอมพิวเตอร์ คือความสามารถใช้ข้อมูลที่เกิดขึ้นในอดีต เพื่อจำลองเหตุการณ์ที่เกิดขึ้นในธรรมชาติ และ คาดการณ์ล่วงหน้าในสถานการณ์เฉพาะที่กำหนดให้ โดยก่อนหน้าที่จะมีคอมพิวเตอร์มาช่วยจำลองเหตุการณ์ นักวิทยาศาสตร์ได้พึ่งแต่ใช้ทฤษฎีต่าง ๆ และใช้การทดลองในห้องทดลองเพื่อพิสูจน์ทฤษฎีเหล่านั้น ทำให้ต้องใช้เวลานานและมีต้นทุนมาก แต่ในปัจจุบัน เราสามารถใช้ทฤษฎีและกฎทางคณิตศาสตร์เหล่านี้มาเป็นหลักในการจำลองเหตุการณ์ด้วยคอมพิวเตอร์ เพื่อแก้ปัญหาที่ไม่เคยถูกค้นพบในห้องทดลองธรรมดาได้เร็วขึ้น และ มีค่าใช้จ่ายถูกลง เช่น การจำลองการเปลี่ยนแปลงของชั้นบรรยากาศของโลกทั้งใบ

ปัจจุบันนี้ แวดวงวิทยาศาสตร์ได้ใช้การจำลองเหตุการณ์ด้วยคอมพิวเตอร์เป็นแกนหลักของโครงการใหญ่หลายโครงการ ดังในรายงาน Simulation-Based Engineering Science Report (2006) ของหน่วยงานในประเทศสหรัฐอเมริกาที่ชื่อ National Science Foundation ซึ่งได้เขียนถึงการจำลองเหตุการณ์ดังนี้:-

"การจำลองเหตุการณ์ทางวิทยาศาสตร์ เป็น แขนงวิชาที่ขาดไม่ได้ ถ้าเราจะให้ชาติของเรายังคงความเป็นผู้นำด้านวิทยาศาสตร์และด้านวิศวกรรม การจำลองเหตุการณ์จะเป็นส่วนสำคัญที่จะทำให้ การแพทย์ชีวภาพ อุตสาหกรรมการผลิตนาโน ความปลอดภัยของแผ่นดินแม่ ไมโครอิเล็กทรอนิกส์ วิทยาศาสตร์พลังงานและสิ่งแวดล้อม วัสดุศาสตร์ และ การพัฒนาผลิตภัณฑ์ ก้าวหน้าขึ้น โดยมีหลักฐานมากมายที่ระบุได้ว่า พัฒนาการในแขนงวิชาใหม่เหล่านี้สามารถสร้างผลกระทบอย่างมีนัยสำคัญกับมนุษย์ในแทบทุกด้าน"

เพื่อให้การจำลองธรรมชาติเหมือนจริง เราจึงต้องใส่ความไม่มีแบบแผน (Randomness) เข้าไปในทุก ๆ การจำลองเหตุการณ์ด้วย เนื่องจากความไม่มีแบบแผนนี้เอง มันจึงเป็นไปได้ที่เราจะเชื่อผลลัพธ์จากการจำลองเพียงแค่ครั้งเดียว เราต้องทำการจำลองหลาย ๆ ครั้งเพื่อหาผลลัพธ์เฉลี่ย (ตามกฎว่าด้วยเลขจำนวนมาก (Law of large numbers) ยิ่งเราจำลองมากครั้งเท่าไร ค่าเฉลี่ยของผลลัพธ์ก็ยิ่งใกล้ค่าที่ถูกต้องมากขึ้นเท่านั้น) โชคดีที่เรามี Supercomputer ขนาดใหญ่ ทำให้เราสามารถจำลองเหตุการณ์ที่มีการประมวลผลแบบโมโหฬารหลาย ๆ ครั้งเสร็จสิ้นในเวลาอันรวดเร็ว

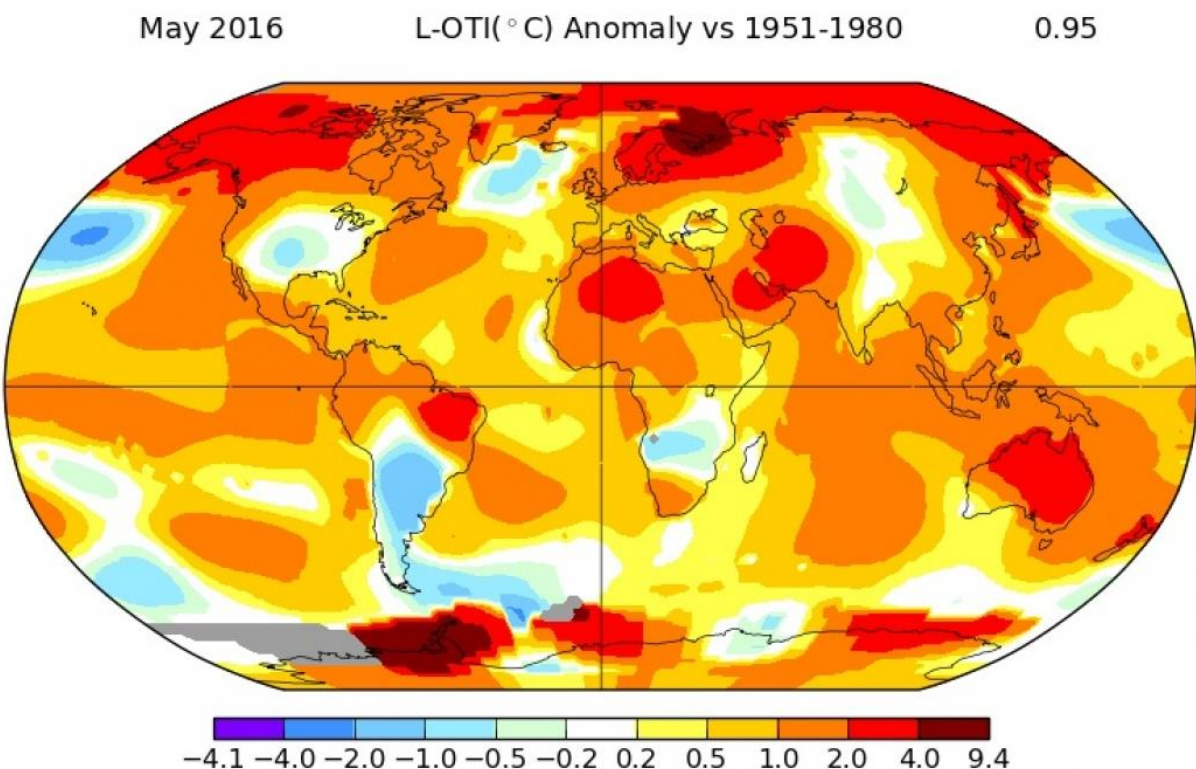
ตัวอย่างของการจำลองเหตุการณ์ด้วยคอมพิวเตอร์

การจำลองเหตุการณ์ได้มีบทบาทสำคัญกับการวิจัยระดับแนวหน้าทางวิทยาศาสตร์ในปัจจุบัน ในบทนี้ เราจะกล่าวถึงโครงการสำคัญของโลก 6 โครงการที่ใช้การจำลองเหตุการณ์ด้วยคอมพิวเตอร์

การเปลี่ยนแปลงในชั้นบรรยากาศ / ภาวะโลกร้อน

ตัวอย่างแรกของการจำลองเหตุการณ์ธรรมชาติของโลก คือ การจำลองการเปลี่ยนแปลงของอุณหภูมิในชั้นบรรยากาศของโลกทั้งใบ คอมพิวเตอร์มีส่วนช่วยอย่างมาก เพราะ เป็นไปไม่ได้ที่มนุษย์จะสามารถเก็บข้อมูลจริงจากชั้นบรรยากาศของโลกทั้งใบ (รูปที่ 11-1 แสดงอุณหภูมิของผิวโลกในเดือนพฤษภาคม 2559 ซึ่งเป็นเดือนที่ร้อนเป็นสถิติสูงสุด ซึ่งการสร้างรูปแบบที่อุณหภูมิต้องอาศัยขั้นตอนวิธี (Algorithm) สำหรับการประมาณอุณหภูมิ ซึ่งถูกนำเสนอครั้งแรกในบทความวิชาการในปี 1981 ของ James Hansen)

จากตัวอย่าง เราจะเห็นได้ว่า คอมพิวเตอร์มีส่วนช่วยอย่างมากในการจำลองเหตุการณ์ โดยเฉพาะในสถานการณ์ที่ข้อมูลมีขนาดใหญ่ และการจำลองต้องคำนึงถึงหลายปัจจัย แม้ว่า การจำลองเหตุการณ์จะมีข้อจำกัดที่ว่า เราต้องมีความเข้าใจของปัญหาย่อย ๆ (Subproblems) เป็นอย่างดีก่อน (เช่น เราจำเป็นต้องทราบ การหักเหของแสงอาทิตย์เมื่อมันตกกระทบกับก้อนเมฆ อากาศที่มีความหนาแน่นสูง หรือ แม้แต่ตกกระทบกับแผ่นดิน) แต่บางครั้งมันก็เป็นวิธีที่ดีที่สุดที่เรา มี เนื่องจากการทดลองวิธีแก้ปัญหาลาย ๆ วิธีในสถานที่จริงแทบเป็นไปไม่ได้



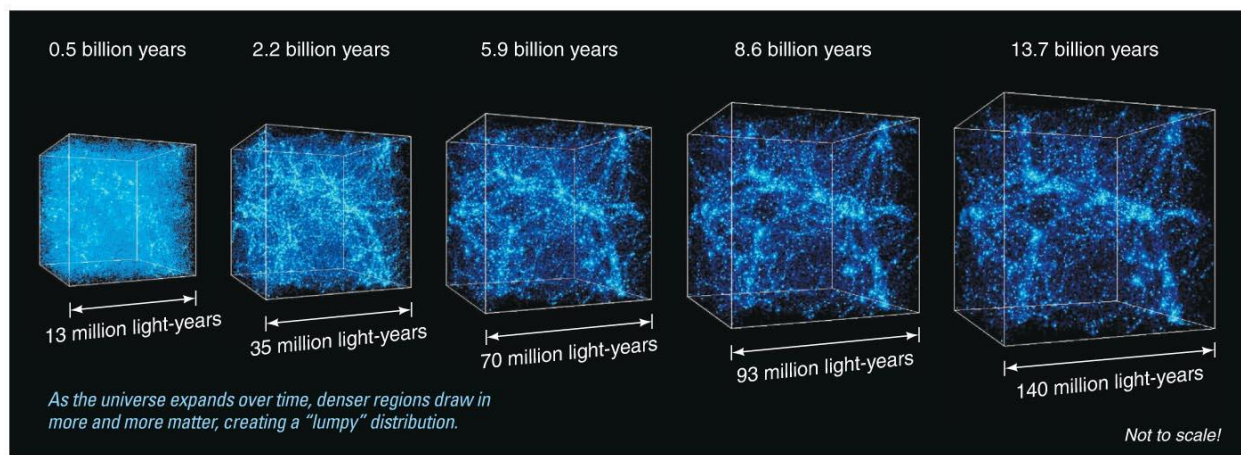
รูปที่ 11-1 อุณหภูมิของผิวโลกในเดือนพฤษภาคม 2559 (อ้างอิง:http://data.giss.nasa.gov/tmp/gistemp/NMAPS/tmp_GHCN_GISS_ERSSTv4_1200km_Anom5_2016_2016_1951_1980_100_180_90_0_2_/amaps.png)

ทฤษฎีการระเบิดครั้งใหญ่ (Big bang theory) / การกำเนิดของโลก

เมื่อมองบนท้องฟ้า คนส่วนใหญ่จะตั้งคำถามทำนองเดียวกันว่า “ท้องฟ้ามีดาวที่ดวงกันนะ?” นักฟิสิกส์จำนวนมากพยายามแก้ปัญหาอันเดียวกันนี้

ถ้าเป็นเมื่อก่อน การหาคำตอบลักษณะนี้เป็นไปไม่ได้เลย แต่ในปัจจุบัน เครื่องมือทางวิทยาศาสตร์ได้ก้าวหน้าไปมาก ดาวเทียมหลายดวงและกล้องโทรทรรศน์อวกาศฮับเบิล (Hubble Space Telescope) ช่วยทำให้เราเข้าใจจักรวาลและจุดกำเนิดของมันมากขึ้น

นักฟิสิกส์หลาย ๆ คนที่คาดเดาว่า การที่จักรวาลได้กลายมาเป็นรูปร่างแบบที่มันเป็นนี้ อย่างไรก็ตามก็ต้องมีอนุภาคหรือสสารบางอย่างที่เราไม่เคยรู้จักมาก่อน (และ ที่ตาคนก็มองไม่เห็น!!) ในจักรวาล ซึ่งเป็นเรื่องยากมากสำหรับนักสังเกตการณ์อวกาศที่จะพิสูจน์ความคิดดังกล่าว แต่หากเรานำเอาการจำลองเหตุการณ์เข้ามาประยุกต์ใช้ เราก็จะพอคาดเดาคำตอบโดยง่ายว่า อนุภาคหรือสสารนั้นมีอยู่จริง นอกจากนั้นแล้วนักดาราศาสตร์ฟิสิกส์ยังสามารถคาดเดาจำนวนของอนุภาคหรือสสารนั้นได้อีกด้วย โครงการนี้แสดงให้เห็นอีกครั้งหนึ่งแล้วว่า การรวบรวมหลักฐานจากที่เราได้สังเกตมา แล้วมาใช้ร่วมกับทฤษฎีทางฟิสิกส์ทำให้เราสามารถจำลองได้ว่า จักรวาลของเรามีหน้าตาอย่างไร มีหรือไม่มีสสารที่ตามองไม่เห็น รูป 11-2 แสดงการใช้คอมพิวเตอร์จำลองการขยายตัวของโครงสร้างของจักรวาลเมื่อเวลาผ่านไป 13.7 ล้านปีแสง



รูปที่ 11-2 การจำลองการขยายตัวของจักรวาลเมื่อเวลาผ่านไปด้วยคอมพิวเตอร์ (อ้างอิง: รูปที่ 18.14 ในหนังสือ The Essential Cosmic Perspective, 7th edition)

การเกิดของสิ่งมีชีวิตในโลก

จุดกำเนิดของสิ่งมีชีวิตบนโลกเป็นอีกหนึ่งประเด็นคำถามที่สำคัญสำหรับหลาย ๆ คน และ สำหรับ นักวิทยาศาสตร์แล้ว ปัญหานี้เป็นปัญหาที่น่าฉงนและยากที่จะทำการทดลองในห้องปฏิบัติการ เพราะการที่จะทำให้ห้องทดลองเป็นเหมือนสภาวะเมื่อโลกมีสิ่งมีชีวิตเกิดขึ้นใหม่ ๆ เป็นเรื่องยากมาก

เริ่มแรกพวกเขาต้องตอบโจทย์พื้นฐานให้ได้ก่อนว่า อะไรคือจุดกำเนิดของ DNA RNA และ กรดอะมิโนตัวแรกสุด เมื่อไม่กี่ปีที่ผ่านมา นักวิทยาศาสตร์ได้ใช้เครื่องซูเปอร์คอมพิวเตอร์เพื่อทดสอบทฤษฎีบางทฤษฎีที่เกี่ยวกับเรื่องนี้ เช่น Thyveetil [2008] ใช้คอมพิวเตอร์จำลองคุณลักษณะของสสารและความเสถียรของโครงสร้างของเลเยอร์ดับเบิลไฮดรอกไซด์ที่มี DNA แทรกกระหว่างชั้น ซึ่งก็เป็นอีกครั้งที่การจำลองเหตุการณ์ได้ถูกประยุกต์ใช้เพื่อแก้ปัญหาทางวิทยาศาสตร์ เนื่องจากมันง่ายกว่า (และถูกกว่า) การสร้างสภาวะของโลกให้เหมือนยุคต้นกำเนิดสิ่งมีชีวิตมาก

การชนกันของทางช้างเผือก

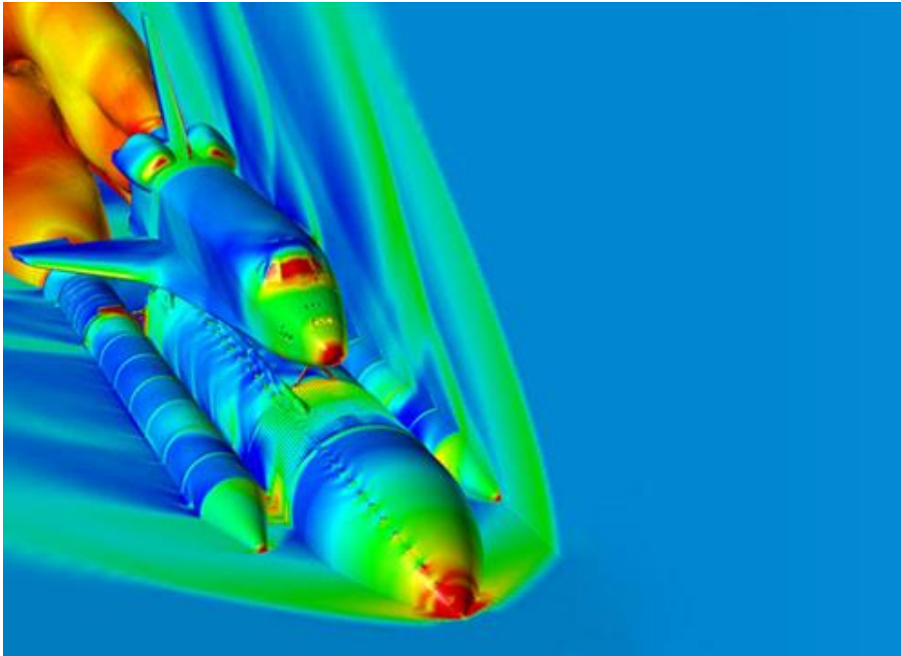
บางการทดลองก็อันตรายเกินไป เช่น การผสมสารเคมีอาจทำให้เกิดไฟลุกไหม้ การทดสอบอาวุธ (เช่นระเบิดนิวเคลียร์) ก็เสี่ยงต่อชีวิตและทรัพย์สินของคนจำนวนมาก ไม่ต้องพูดถึงการทดลองให้กาแล็กซีมาชนกันจริง ๆ เราก็แทบไม่สามารถจินตนาการได้ว่า การทดลองให้ทางช้างเผือกมาชนกันนั้นจะอันตรายขนาดไหน จะทำการทดลองได้ยากขนาดไหน เพราะเราแทบจะควบคุมอะไรไม่ได้เลย

แม้มันจะยาก แต่บางทีเราก็ต้องศึกษาการชนกันของกาแล็กซีให้ได้ เพราะกาแล็กซีของเรา (หรือ ทางช้างเผือก) อาจจะชน กาแล็กซีชื่อ Andromeda ในเวลา 3 ล้านปีแสงก็ได้ ถ้าเป็นอย่างนั้นจริง อะไรจะเกิดขึ้นบ้างกับสิ่งมีชีวิตที่อาศัยในกาแล็กซีทั้งสอง กาแล็กซีทั้งสองจะชนแล้วแยกออกจากกัน หรือ จะรวมกันเป็นหนึ่งเดียว ซึ่งแน่นอน เราสามารถใช้คอมพิวเตอร์ช่วยจำลองทั้งสองเหตุการณ์ได้จากข้อมูลของกล้องโทรทรรศน์อวกาศฮับเบิล (นิสิตสามารถดูคลิปวิดีโอการจำลองการชนกันของกาแล็กซีได้จาก URL นี้ : <http://www.space.com/15947-milky-andromeda-galaxies-collision-simulated-video.html>)

การออกแบบกระสวยอวกาศ เครื่องบินเจ็ต และ รถยนต์

ปัจจุบัน เราสามารถจำลอง แรงต้านของลม การเปลี่ยนแปลงของอุณหภูมิ ความแปรปรวนของแรงดัน และ ปัจจัยต่าง ๆ อีกมากมายได้ใกล้เคียงกับของจริงมากยิ่งขึ้น (รูปที่ 11-3 แสดงให้เห็นถึงการจำลองแรงกดดันอากาศต่อตัวกระสวยอวกาศ (และ เส้นทางไหลรอบ ๆ ตัวกระสวย) เมื่อถูกยิงออกจากฐาน) และ เรายังสามารถคำนวณค่าต่าง ๆ ภายในหน่วยเวลาเป็นนาที่ แทนที่จะเป็นวัน ๆ หรือ เป็นสัปดาห์ ๆ นอกจากนี้ เรายังสามารถทดลองออกแบบได้หลาย ๆ แบบก่อนที่จะนำแบบที่เหมาะสมที่สุดไปผลิตจริง การจำลองผลกระทบกับแบบที่เราออกช่วยให้เราทราบว่า แบบนั้น ๆ คุ้มค่ากับการผลิตหรือไม่ นั่นคือ การจำลองเหตุการณ์เป็นกุญแจสำคัญช่วยให้ 1) การ

วางแผนและการทดสอบแบบต่าง ๆ สามารถทำได้สำเร็จได้ในระยะเวลาสั้นกว่า และใช้ต้นทุนน้อยกว่าในอดีต 2) ผู้ออกแบบสามารถรู้ได้ว่า ส่วนใดของแบบที่เป็นจุดอ่อนที่ต้องถูกเสริมให้แกร่ง และ 3) การแก้แบบให้เหมาะสมในแบบจำลองนั้นทำได้ง่ายกว่าต้นแบบที่อยู่ในสายการผลิตจริง



รูปที่ 11-3 การจำลองแรงกดดันอากาศต่อตัวกระสวยอวกาศ และ เส้นทางไหลของอากาศรอบ ๆ (http://www.hec.nasa.gov/news/gallery_images/gomez.shuttle.jpg)

โปรแกรมช่วยการจำลองเหตุการณ์ (Scratch)

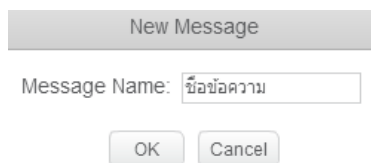
ตัวอย่างของโปรแกรมสำหรับการจำลองเหตุการณ์ในบทนี้ จะกล่าวถึง 1) การจำลองการสนทนาของตัวละครที่มากกว่า 1 ตัวละคร และ 2) การจำลองการเข้าแถวคอยเพื่อซื้อสินค้า

ตัวอย่างที่ 1 การจำลองการสนทนาของตัวละครที่มากกว่า 1 ตัวละคร

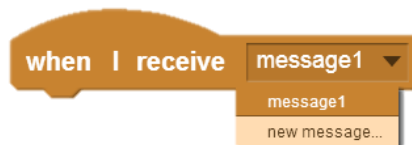
โครงร่างของเรื่อง คือ ตัวละครสนทนากับผู้ใช้เล็กน้อย จากนั้นจะแนะนำวัตถุดิบสำหรับการทำอาหารให้ผู้ใช้ โดยวัตถุดิบแต่ละอย่างนั้นจะปรากฏขึ้นครั้งละ 1 ชิ้นเท่านั้น บล็อกคำสั่งที่สั่งที่จำเป็นสำหรับการสร้างโปรแกรมนี้นี้คือบล็อกคำสั่ง broadcast โดยบล็อกคำสั่งนี้จะอยู่ในหมวดบล็อกคำสั่ง Events (สังเกต กรอบสีแดงในรูปซ้ายของหน้าถัดไป)



broadcast ชื่อข้อความ คือ บล็อกคำสั่งสำหรับส่งค่าหรือส่งสัญญาณไปให้อีกตัวละครหนึ่ง โดยตัวเลือกด้านหลังของบล็อกคำสั่งนี้ คือ ข้อความที่จะต้องสื่อสารกันระหว่างตัวละครสองตัว ผู้ใช้สามารถตั้งข้อความให้แตกต่างกันได้ และ ควรตั้งข้อความให้สื่อความหมายว่า เป็นข้อความจากตัวละครใดไปยังตัวละครใด การตั้งข้อความสำหรับส่งให้ตัวละครนี้สามารถทำได้โดยการกดเลือกที่ตัวเลือกด้านหลังของบล็อกคำสั่ง และเลือก new message... จากนั้นจะปรากฏหน้าต่างเล็ก ๆ บริเวณกลางหน้าจอ ดังนี้



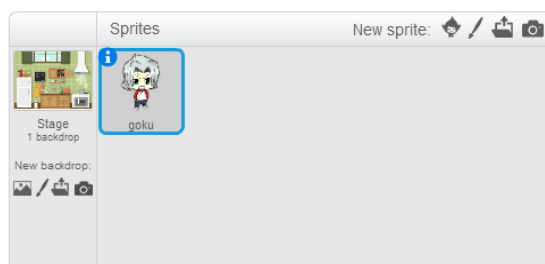
จากนั้น ให้กรอกข้อความที่ต้องการลงในกล่องข้อความ แล้วกดปุ่ม OK



เมื่อมีการส่งข้อความจากตัวละครหนึ่งไปยังอีกตัวละครแล้ว ตัวละครอีกตัวจะต้องมีการรับข้อความหรือสัญญาณนี้ด้วย บล็อกคำสั่งสำหรับรับข้อความและทำงานในส่วนที่เกี่ยวข้องต่อไปคือ บล็อกคำสั่ง when I receive ผู้ใช้สามารถกดตัวเลือกที่อยู่บริเวณด้านหลังของบล็อกคำสั่งเพื่อเลือกข้อความจากข้อความทั้งหมดในชิ้นงาน โดยผู้ใช้ควรเลือกข้อความให้ถูกต้อง ตรงกับตัวละครที่จะได้ส่งข้อความนั้นมา

ตัวอย่างของการใช้บล็อกคำสั่งทั้งรับและส่งข้อความนี้จะแสดงโดยจำลองสถานการณ์การทำอาหารในครัว และผู้ประกอบอาหารจะต้องแนะนำวัตถุดิบสำหรับการทำอาหาร ที่ละวัตถุดิบ ดังนั้น วัตถุดิบแต่ละชิ้นจะต้องปรากฏขึ้นหลังจากที่ตัวละครแนะนำเรียบร้อยแล้ว และจะต้องหายไปก่อนที่จะตัวละครจะแนะนำวัตถุดิบชิ้นใหม่ ซึ่งมีขั้นตอนดังนี้

เริ่มต้นจากการเพิ่มตัวละครหลัก และฉากหลังให้เรียบร้อยแล้ว

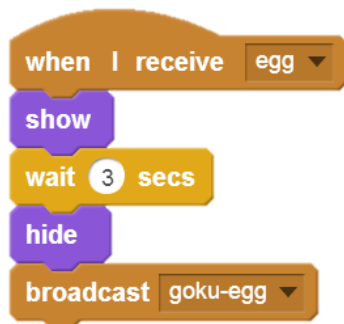
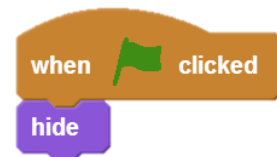


จากนั้นวางบล็อกคำสั่งให้กับตัวละครหัวปลาหมึก (goku) ให้แนะนำตัวเองเล็กน้อย เมื่อแนะนำตัวเองเรียบร้อยแล้วต่อไปจะเป็นการแนะนำวัตถุดิบสำหรับการปรุงอาหาร โดยวัตถุดิบแรกคือไข่ไก่ ผู้ใช้ทำการเพิ่มรูปภาพไข่ไก่เข้ามาให้เรียบร้อยแล้ว (รายละเอียดการเพิ่มรูปภาพอยู่ในบทปฏิบัติการที่ 7) จากนั้นจะเป็นการเริ่มวางบล็อกคำสั่ง broadcast ดังนี้



บล็อกคำสั่ง when (ธงสีเขียว) clicked เป็นบล็อกกระตุ้นคำสั่งให้ตัวละครเริ่มทำงานเมื่อผู้ใช้คลิกธงสีเขียวที่บริเวณด้านบนขวาของส่วนแสดงผลตัวละคร และจากหลัง (🇹🇵) ส่วนบล็อกคำสั่ง broadcast egg เป็นการส่งข้อความ egg ไปยังตัวละครไข่

การวางบล็อกคำสั่งให้กับรูปภาพหรือตัวละครไข่นั้น จะต้องเริ่มวางบล็อกเช่นเดียวกับตัวละครหลัก คือ จะต้องวางบล็อกคำสั่ง when (ธงสีเขียว) clicked จากนั้นวางบล็อกคำสั่ง hide ต่อ เพื่อจะจางให้ตัวละครไข่ยังไม่ปรากฏพร้อมกับ goku แต่จะให้ไข่ปรากฏเมื่อ goku กล่าวถึงเท่านั้น



การทำให้ไข่ปรากฏเมื่อ goku กล่าวถึงทำได้โดยใช้บล็อก when I receive โดยวางบล็อกคำสั่ง ดังนี้

ชุดของบล็อกคำสั่งนี้จะเริ่มต้นด้วยการรับข้อความ egg จาก goku แล้วตัวละครจะแสดงตัวออกมาเป็นเวลา 3 วินาที จากนั้นก็ซ่อนตัวเหมือนเดิม นอกจากนี้ ไข่ไก่อังสามารถส่งข้อความให้ goku ว่า goku-egg เพื่อให้สัญญาณกับ goku ว่า ไข่ไก่ได้แสดงตัวเรียบร้อยแล้ว และ goku สามารถทำงานอื่นๆ ต่อได้



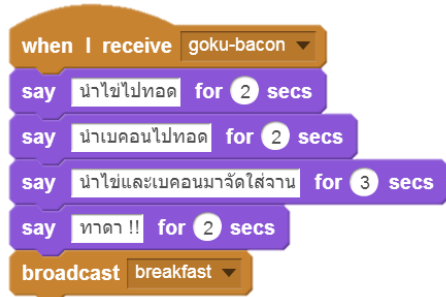
จากนั้น กลับมาที่ goku ซึ่งได้รับข้อความจากตัวละครไข่ และแนะนำตัวละคร (วัตถุดิบ) ต่อไป บล็อกคำสั่ง when I receive ..goku-egg.. เป็นการรับข้อความที่ตัวละครไข่ส่งมานั่นเอง หลังจากรับข้อความแล้ว ตัวละครจะกล่าวต่อว่า “มีเบคอน” และ ส่งข้อความไปให้ตัวละครเบคอน เพื่อให้ตัวละครเบคอนปรากฏขึ้นต่อไป

ฝั่งของตัวละครเบคอนจะต้องวางบล็อกคำสั่งชุดแรกเช่นเดียวกับตัวละครไข่ไก่ คือ จะต้องซ่อนก่อนเมื่อผู้ใช้คลิกปรังสีเขียว และจะปรากฏขึ้นได้เมื่อ goku ส่งข้อความมาให้ บล็อกคำสั่งของตัวละครเบคอนหลังจากได้รับข้อความจาก goku แล้ว เป็นดังนี้



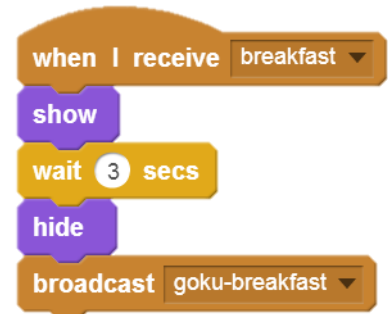
บล็อกคำสั่งแรก ตัวละครเบคอนรับข้อความจาก goku ว่า “bacon” จากนั้นตัวละครเบคอนปรากฏขึ้นเป็นเวลา 3 วินาที และซ่อนเช่นเดิม จากนั้นตัวละครเบคอนจะส่งข้อความกลับไปยัง goku ว่า “goku-bacon” ที่บล็อกคำสั่งสุดท้ายของชุดคำสั่งนี้

จากนั้นจะเป็นบทพูดของ goku เพื่อแนะนำให้นาไข่ไก่และเบคอนลงทอดในกระทะ พร้อมจัดใส่จาน และแสดงรูปภาพของจานอาหารเข้าที่จัดแล้ว โดย goku จะส่งข้อความไปยังตัวละครจานอาหาร ดังนี้

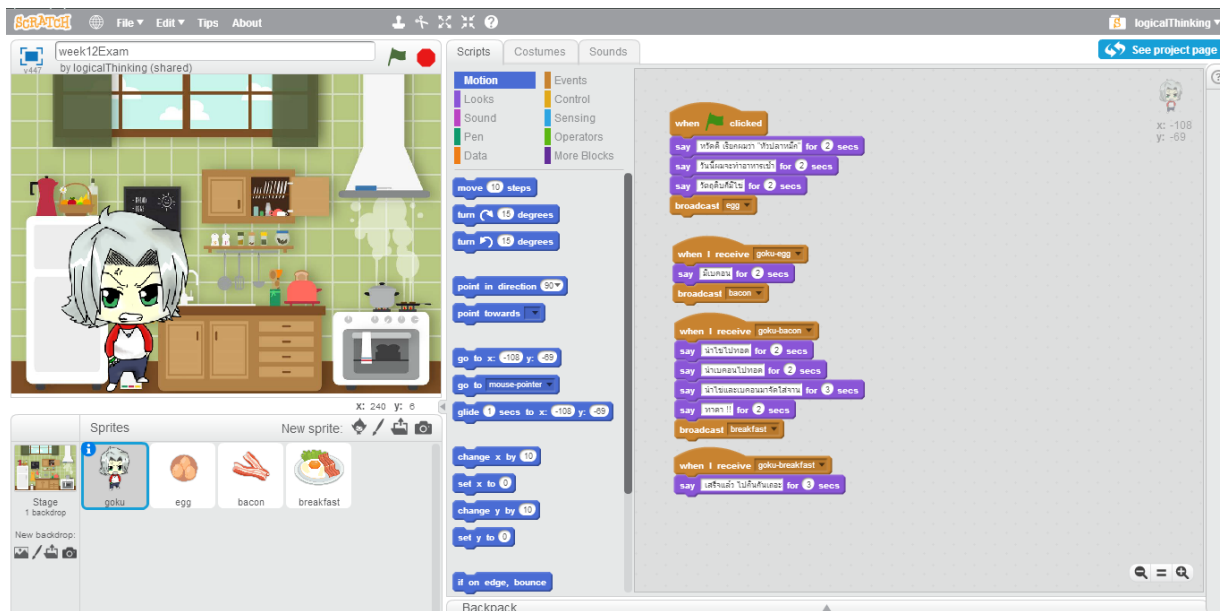


จากชุดบล็อกคำสั่งนี้ บล็อกแรกคือการที่ goku รับค่า “goku-bacon” มาจากตัวละครเบคอน และบล็อกคำสั่งสุดท้ายคือ goku ส่งข้อความ “breakfast” ไปยังตัวละครจานอาหาร

ตัวละครจานอาหารจะต้องถูกซ่อนไว้ก่อนเมื่อผู้ใช้คลิกปรังสีเขียว เช่นเดียวกับตัวละครไข่ไก่และเบคอน เมื่อ goku ส่งข้อความ “breakfast” มาถึงตัวละครจานอาหารแล้วตัวละครจานอาหารจึงปรากฏขึ้นได้โดยวางบล็อกคำสั่ง when I receive breakfast เป็นบล็อกคำสั่งแรก จากนั้นตัวละครจานอาหารจะแสดงเป็นเวลา 3 วินาที และซ่อนตัวเช่นเดิม สุดท้าย จานอาหารส่งข้อความ “goku-breakfast” กลับไปยัง goku เพื่อให้ goku ทำส่วนต่อไป



การรับข้อความสุดท้ายของตัวอย่างนี้คือ goku รับข้อความ “goku-breakfast” จากตัวละครจานอาหาร และกล่าวจบด้วยข้อความว่า “เสร็จแล้ว ไปกินกันเถอะ” เป็นอันจบการทำงานของโปรแกรมตัวอย่างแสดงบทสนทาระหว่างตัวละครตั้งแต่ 2 ตัวละครขึ้นไป สามารถชมตัวอย่างและบล็อกคำสั่งทั้งหมดได้ที่ <https://scratch.mit.edu/projects/112120973/>



ตัวอย่างที่ 2 การจำลองการเข้าแถวคอยเพื่อซื้อสินค้าที่ร้านขายหมูปัง

ตัวอย่างต่อไปจะเป็นการแสดงการเข้าแถวคอยเพื่อซื้อสินค้าที่ร้านขายหมูปัง โดยจะใช้บล็อกคำสั่งเพิ่มเติมคือบล็อกคำสั่งเกี่ยวกับการโคลนตัวละคร และบล็อกคำสั่งอื่น ๆ ดังนี้

next costume

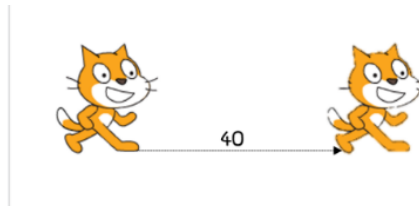


move 40 steps

บล็อกคำสั่ง next costume เป็นบล็อกคำสั่งสำหรับสั่งให้ตัวละครแสดงผลในรูปแบบที่แตกต่างออกไป โดยในตัวอย่างที่ 2 นี้จะใช้สำหรับการเพิ่มความเหมือนจริงสำหรับการเคลื่อนไหวของตัวละครลูกค้าที่จะต้องเดินเข้ามาต่อคิวเพื่อซื้อหมูปัง

ตัวอย่างเช่นตัวละครแมวจะมีชุด 2 รูปแบบ (costume1 และ costume2) แต่ไม่ว่าทุกตัวละครจะมีรูปแบบที่หลากหลาย บางตัวละครจะมีเพียงรูปแบบเดียวเท่านั้น สามารถตรวจสอบรูปแบบของตัวละครที่เลือกมาโดยการคลิกที่ตัวละครนั้นคลิกเลือก tab Costumes บริเวณด้านบนของหมวดบล็อกคำสั่ง

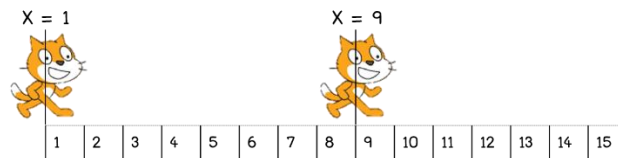
บล็อกคำสั่ง move.....steps เป็นบล็อกคำสั่งสำหรับให้ตัวละครเคลื่อนที่ โดยมีการระบุจำนวนหน่วยที่ต้องการให้ตัวละครเคลื่อนที่ จากตัวอย่าง move 40 steps จะมีผลให้ตัวละครเคลื่อนที่จากตำแหน่งเดิมไป 40 หน่วย ตามแนวแกน x



จากภาพ เมื่อเริ่มต้นตัวละครจะอยู่ที่ตำแหน่งทางด้านซ้าย และหลังจากวางบล็อกคำสั่ง move 40 steps แล้ว ตัวละครจะเคลื่อนไปยังตำแหน่งด้านขวา หรือตำแหน่งปลายทาง โดยมีความห่างจากตำแหน่งเดิม 40 หน่วย และในตัวอย่างที่ 2 นี้จะใช้สำหรับการเคลื่อนที่ของลูกค้า ที่จะต้องเดินจากขอบของฉากมาเรื่อย ๆ จนถึงตัวแม่ค้า ทำการเจรจากับแม่ค้า และเคลื่อนที่ออกจากฉากเมื่อเจรจากับแม่ค้าเรียบร้อยแล้ว

x position

บล็อกคำสั่ง x position บล็อกคำสั่งนี้สามารถเรียกดูค่าตำแหน่ง x ปัจจุบันของตัวละครได้ อาจใช้สำหรับตั้งค่าตำแหน่งเริ่มต้นให้กับตัวละครเมื่อผู้ใช้กดรูปรองสีเขียว



จากภาพ ถ้าวางบล็อกคำสั่ง x position ให้กับตัวละครก่อนก้าวเดิน จะทราบตำแหน่งของตัวละครตัวแรก (คือ 1) แต่ถ้าวางบล็อกคำสั่ง x position ให้กับตัวละครหลังจากก้าวเดินแล้ว จะได้ค่าตำแหน่งเป็น 9 หรือ บล็อกคำสั่งนี้สามารถถูกใช้ระบุให้ตัวละครมาอยู่ที่ตำแหน่งที่ระบุก็ได้ บล็อกคำสั่งนี้จะถูกใช้งานสำหรับเรียกดูตำแหน่งปัจจุบันของตัวละครลูกค้า และนำไปใช้ต่อในเงื่อนไขที่จะเลื่อนคิวของลูกค้าจากคิวด้านหลัง มาอยู่ในคิวด้านหน้า

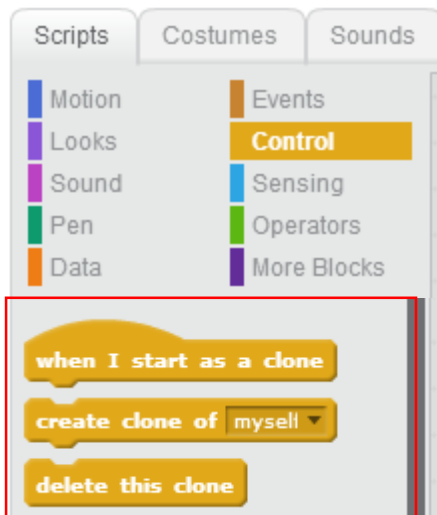
point in direction 95

บล็อกคำสั่ง point in direction ..มุม.. บล็อกคำสั่งนี้เป็นการกำหนดทิศทางการเคลื่อนที่ของตัวละคร โดยกำหนดเป็นองศา จากตัวอย่าง บล็อกคำสั่งจะกำหนดทิศทางการเคลื่อนที่ที่เป็น 95 องศา โดยจะเริ่มนับ 0 องศาที่จุดกึ่งกลางด้านบนของตัวละครหมุนตามเข็มนาฬิกาไป 95 องศา



จากภาพ เมื่อวางบล็อกคำสั่ง point in direction 95 ในกับตัวละครแล้ว ตัวละครจะเริ่มเดินจากซ้ายมือ แล้วเคลื่อนที่ไปยังตำแหน่งผลลัพธ์ (ตำแหน่งขวาล่าง) นั่นเอง บล็อกคำสั่งนี้จะถูกใช้กับตัวละครลูกค้า ช่วงหลังจากเจรจากับแม่ค้าเรียบร้อยแล้ว และจะต้องเดินเบี่ยงร้านค้า และเดินออกจากฉากไปในที่สุด

บล็อกคำสั่งสำหรับการโคลนตัวละครจะอยู่ในหมวดบล็อกคำสั่ง Control



ประกอบด้วยบล็อกคำสั่ง when I start as a clone บล็อกคำสั่ง create clone of และบล็อกคำสั่ง delete this clone การโคลนตัวละครนี้จะถูกใช้กับตัวละครลูกค้า เพื่อเพิ่มจำนวนตัวละครลูกค้าตามอัตราที่ผู้ใช้กำหนด รายละเอียดของแต่ละบล็อกคำสั่ง คือ

when I start as a clone

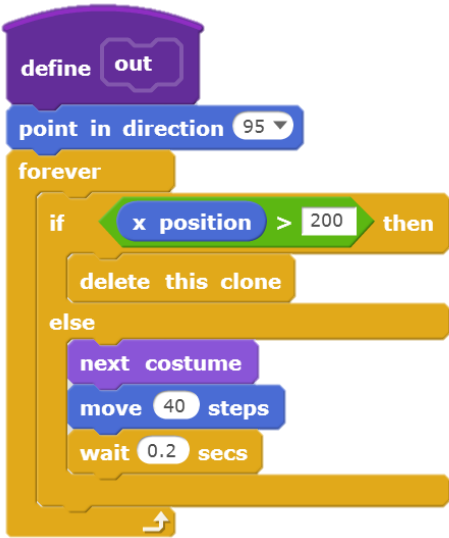
ตัวละครที่ถูกโคลนจะเริ่มทำงานภายใต้บล็อกคำสั่งนี้

create clone of myself

ตัวละครที่ถูกกระตุ้นด้านหลังของ

บล็อกคำสั่ง create clone of ...myself.. คือตัวละครที่จะถูกโคลน และ บล็อกคำสั่ง delete this clone ใช้สำหรับลบตัวละครที่ถูกโคลน

ตัวอย่างการเข้าแถวคอยเพื่อซื้อหมูปิ้ง เริ่มจากการนำเข้าตัวละครทั้งลูกค้า และผู้ค้า ฉาก ร้านหมูปิ้ง และหมูปิ้งก่อน จากนั้นเริ่มวางบล็อกคำสั่งโดยมีลำดับดังนี้



ของฟังก์ชัน out

บล็อก define ฟังก์ชัน out ใช้สร้างชุดคำสั่ง มีรายละเอียด ดังนี้

บล็อกคำสั่ง point in direction __ ใช้สำหรับ ระบุให้ตัวละครลูกค้า เดินลงจากเดิม 5 องศา จากนั้นการตรวจสอบว่า ถ้า ตำแหน่ง x มากกว่า 200 ตัวละครลูกค้าที่โคลนมาจะถูกลบ หมายถึงตัวละครลูกค้าเดินออกนอกบริเวณเซตร้านค้าแล้ว จะต้องหายออกไปจากฉาก ถ้าตำแหน่ง x น้อยกว่า 200 ตัวละครลูกค้า จะเปลี่ยนท่าทาง และเคลื่อนที่ต่อไป 40 หน่วยและรอ 2 วินาที เพื่อสร้างความเสมือนจริงให้กับตัวละครลูกค้าให้มีท่าทางในการก้าวเดินเพื่อไปยังตำแหน่งที่กำหนด โดยการสลับรูปแบบของตัวละครลูกค้า เป็นอันจบการทำงาน



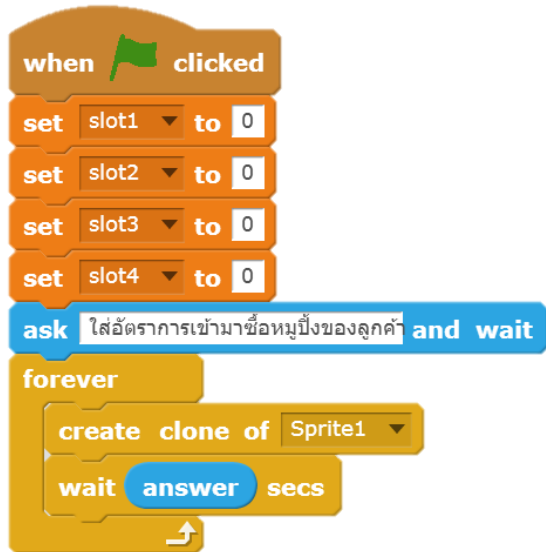
บล็อกคำสั่ง broadcast..message1..and wait ไปยังตัวละครแม่ค้า เพื่อให้ตัวละครแม่ค้าสนทนาตอบโต้กลับ เมื่อเรียบร้อยแล้วตัวละครลูกค้าจะกล่าวว่า “ขอบคุณครับ” ด้วยบล็อกคำสั่ง say..ขอบคุณครับ..for 1 secs

บล็อก define ฟังก์ชัน buy ใช้สร้างชุดคำสั่ง เพื่อสร้างบทสนทาระหว่างตัวละครลูกค้าที่ถูกโคลนใด ๆ กับผู้ค้าหมูปัง ภายในบล็อกคำสั่ง define ฟังก์ชัน buy นี้ จะเริ่มต้นด้วยบล็อกคำสั่ง say..ซื้อหมูปังพร้อมข้าวเหนียวครับ..for 2 secs เป็นบทสนทนาที่ลูกค้ากล่าวกับแม่ค้า เมื่อตัวละครกล่าวเรียบร้อยแล้วจะส่งข้อความโดย

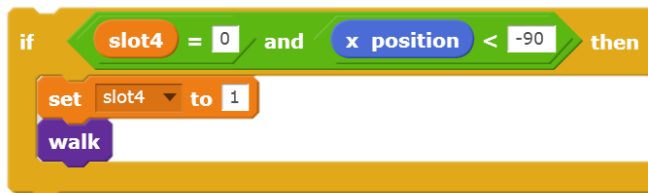


บล็อก define ฟังก์ชัน walk ใช้สร้างชุดคำสั่งเพื่อนำไปใช้กับตัวละครลูกค้าเพื่อให้ตัวละครลูกค้ามีการเคลื่อนไหว หรือการเดินที่มีความเสมือนจริงมากขึ้นวนการเข้าแถวคอยเพื่อซื้อหมูปัง

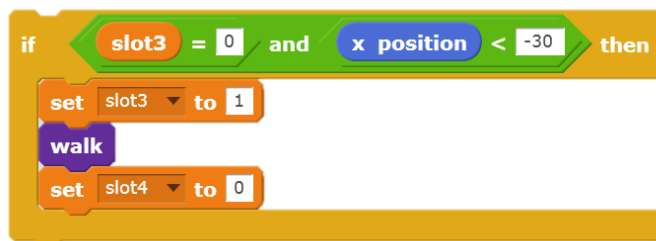
ส่วนของโปรแกรมหลักนอกเหนือจากฟังก์ชันดังกล่าว จะมีการวางตำแหน่งของการเข้าแถวคอยโดยสมมุติให้มีลำดับของแถวคอยเป็นลำดับที่ 1, 2, 3, และ 4



ทำงานแบบวนซ้ำตลอด (forever) โดยคำสั่งภายในบล็อกจะถูกทำซ้ำตลอดเวลา เพื่อโคลนตัวละคร Sprite1 และ รอเป็นจำนวนวินาที ตามอัตราการโคลนตัวละคร (wait answer secs) ที่ผู้ใช้จะกรอกเข้ามาด้วย



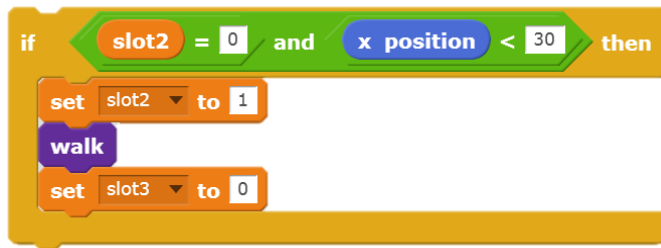
ลูกค้า แล้วให้ลูกค้าเดินเข้า (function walk)



บล็อกคำสั่งชุดนี้เป็นการตรวจสอบตำแหน่ง4 (ตำแหน่งซ้ายสุดภายในแถวคอย) ของลูกค้าคิวที่ 4 ว่าว่าง และ ตำแหน่งตัวละครอยู่ที่ตำแหน่ง $x < -90$ หรือไม่ ถ้าเงื่อนไขทั้งสองเป็นจริง ให้จองตำแหน่งคิวที่ 4 ให้ตัวละคร

มายังตำแหน่งคิวที่ 4

ตรวจสอบตำแหน่งคิวที่ 3 ว่าว่าง และ ตำแหน่งตัวละครอยู่ที่ตำแหน่ง $x < -30$ หรือไม่ ถ้าเงื่อนไขทั้งสองเป็นจริง ให้จองตำแหน่งคิวที่ 3 ให้ตัวละครลูกค้า ตัวละครลูกค้าเดินเข้ามา ทำให้ตำแหน่งคิวที่ 4 ว่าง

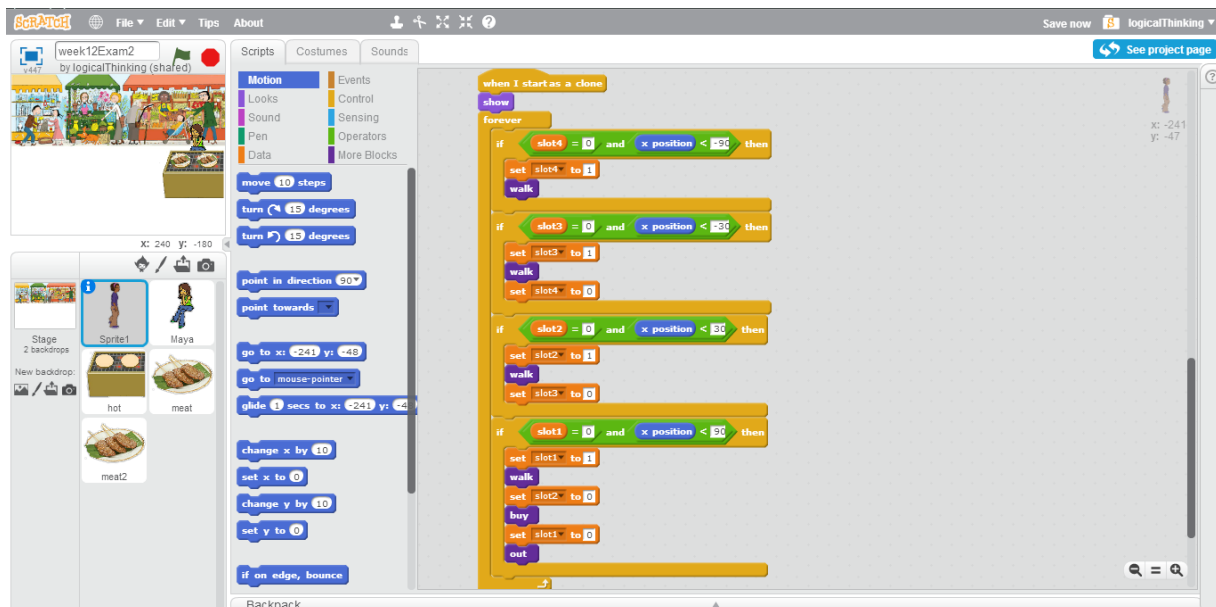


ตรวจสอบตำแหน่งคิวที่ 2 ว่าว่าง และ ตำแหน่งตัวละครอยู่ที่ตำแหน่ง $x < 30$ หรือไม่ ถ้าเงื่อนไขทั้งสองเป็นจริง ให้จองตำแหน่งคิวที่ 2 ให้ตัวละครลูกค้า ตัวละครลูกค้าเดินเข้ามา ทำให้ตำแหน่งคิวที่ 3 ว่าง

ตรวจสอบตำแหน่งคิวที่ 1 ว่าว่าง และ ตำแหน่งตัวละครลูกค้าอยู่ที่ตำแหน่ง $x < 90$ หรือไม่ ถ้าเงื่อนไขทั้งสองเป็นจริง ให้จองตำแหน่งคิวที่ 1 ให้ตัวละครลูกค้า ตัวละครลูกค้าเดินเข้ามา (function walk) ทำให้ตำแหน่งคิวที่ 2 ว่าง ตัวละครลูกค้าที่เดินเข้ามาสนทนากับแม่ค้า (function buy) ทำให้ตำแหน่งคิวที่ 1 ว่างเรียบร้อยแล้วตัวละครลูกค้าเดินออกจากฉาก (function out)



นิสิตสามารถศึกษาและทำความเข้าใจเพิ่มเติมจากตัวอย่างการเข้าแถวคอยนี้ได้ที่ <https://scratch.mit.edu/projects/112247920/>



- http://bjc.berkeley.edu/bjc-r/topic/topic.html?topic=berkeley_bjc/apps/simulation-in-science.topic&course=berkeley_bjc.html&noassignment
- Hansen, J., D. Johnson, A. Lacis, S. Lebedeff, P. Lee, D. Rind, and G. Russell, 1981: Climate impact of increasing atmospheric carbon dioxide. *Science*, 213, 957-966, doi:10.1126/science.213.4511.957
- BENNETT / DONAHUE / SCHNEIDER / VOIT *The Essential Cosmic Perspective*, 7/e ISBN-10: 0321927842 | ISBN-13: 9780321927842
- Computer Simulation Study of the Structural Stability and Materials Properties of DNA-Intercalated Layered Double Hydroxides' by Mary-Ann Thyveetil, Peter Coveney, H. Chris Greenwell and James Suter, is published online in the *Journal of the American Chemical Society* on Tuesday 18 March 2008.