

ใบความรู้ที่ 1 เรื่อง การเขียนโปรแกรมเบื้องต้น

โปรแกรมคอมพิวเตอร์ (Computer Programming) คือ ชุดคำสั่งที่สั่งให้คอมพิวเตอร์สามารถทำงานได้ตรงตามความต้องการและความถูกต้อง เช่น โปรแกรมคอมพิวเตอร์สำหรับสั่งให้คอมพิวเตอร์พิมพ์เอกสาร โปรแกรมสำหรับวาดภาพ เป็นต้น

การเขียนโปรแกรม (Programming) คือ การเขียนชุดคำสั่งด้วยภาษาโปรแกรมที่สั่งให้คอมพิวเตอร์สามารถทำงานได้ตรงตามความต้องการ และสามารถทำงานได้อย่างถูกต้อง ซึ่งเป็นการกำหนดขั้นตอนให้กับคอมพิวเตอร์ทำงานตามลำดับและรูปแบบที่กำหนดไว้

การเขียนโปรแกรมคอมพิวเตอร์ (Computer programming) หรือเรียกให้สั้นลงว่า **การเขียนโปรแกรม (อังกฤษ: Programming)** หรือ **การเขียนโค้ด (Coding)** เป็นขั้นตอนการเขียนทดสอบ และดูแลซอร์สโค้ดของโปรแกรมคอมพิวเตอร์ ซึ่งซอร์สโค้ดนั้นจะเขียนด้วยภาษา **โปรแกรม** ขั้นตอนการเขียนโปรแกรมต้องการความรู้ในหลายด้านด้วยกัน เกี่ยวกับโปรแกรมที่ต้องการจะเขียน และขั้นตอนวิธีที่จะใช้ ซึ่งในวิศวกรรมซอฟต์แวร์นั้น การเขียนโปรแกรมถือเป็นเพียงขั้นหนึ่งในวงจรของการพัฒนาซอฟต์แวร์

การพัฒนาโปรแกรม

การพัฒนาโปรแกรม หมายถึง การเขียนคำสั่งให้คอมพิวเตอร์ประมวลผลข้อมูล มีการใช้ตัวแปร (Variable) เก็บข้อมูล และลำดับการทำงานของโปรแกรมเป็นไปอย่างชัดเจน อาจมีการทำงานตามลำดับ การทำงานแบบทางเลือก และการทำงานแบบวนซ้ำ การพัฒนาโปรแกรมคอมพิวเตอร์ในยุคที่มีภาษาระดับสูง เช่น ภาษาโลโก้ ภาษาซี โดยใช้หลักการเขียนโปรแกรมแบบโครงสร้าง (structural programming) โปรแกรมแบบโครงสร้างสามารถเริ่มต้นเขียนและทำความเข้าใจได้ง่าย แต่เมื่อโปรแกรม มีขนาดใหญ่ขึ้นการเขียนโปรแกรมจะมีขั้นตอนที่ซับซ้อนและใช้เวลาในการพัฒนาโปรแกรมมากขึ้น ในการสั่งงานให้คอมพิวเตอร์ทำตามความต้องการ จำเป็นที่ต้องเขียนโปรแกรมคอมพิวเตอร์เพื่อให้คอมพิวเตอร์ปฏิบัติตาม ดังนั้นโปรแกรมคอมพิวเตอร์ คือ กลุ่มของคำสั่งโดยละเอียดที่สั่งให้เครื่องคอมพิวเตอร์ทำงาน

ขั้นตอนการพัฒนาโปรแกรม

ขั้นตอนการพัฒนาโปรแกรมประกอบด้วย

1. การวิเคราะห์ปัญหา
2. การออกแบบโปรแกรม
3. การเขียนโปรแกรมด้วยภาษาคอมพิวเตอร์
4. การทดสอบและแก้ไขโปรแกรม
5. การทำเอกสารประกอบโปรแกรม
6. การบำรุงรักษาโปรแกรม

1. การวิเคราะห์ปัญหา

การวิเคราะห์ปัญหา ประกอบด้วยขั้นตอนต่างๆ ดังนี้

1. กำหนดวัตถุประสงค์ของงาน เพื่อพิจารณาว่าโปรแกรมต้องทำการประมวลผลอะไรบ้าง
2. พิจารณาข้อมูลนำเข้า เพื่อให้ทราบว่าต้องนำข้อมูลอะไรเข้าคอมพิวเตอร์ ข้อมูลมีคุณสมบัติเป็นอย่างไร ตลอดจนถึงลักษณะและรูปแบบของข้อมูลที่จะนำเข้า
3. พิจารณาการประมวลผล เพื่อให้ทราบว่าโปรแกรมมีขั้นตอนการประมวลผลอย่างไรและมีเงื่อนไขการประมวลผลอะไรบ้าง
4. พิจารณาข้อสนเทศนำออก เพื่อให้ทราบว่า มีข้อสนเทศอะไรที่จะแสดง ตลอดจนรูปแบบและสื่อที่จะใช้ในการแสดงผล

2. การออกแบบโปรแกรม

การออกแบบขั้นตอนการทำงานของโปรแกรมเป็นขั้นตอนที่ใช้เป็นแนวทางในการลงรหัสโปรแกรม ผู้ออกแบบขั้นตอนการทำงานของโปรแกรมอาจใช้เครื่องมือต่างๆ ช่วยในการออกแบบ อาทิ เช่น คำสั่งจำลอง (Pseudocode) หรือ ผังงาน (Flow chart) การออกแบบโปรแกรมนั้นไม่ต้องพะวงกับรูปแบบคำสั่งภาษาคอมพิวเตอร์ แต่ให้มุ่งความสนใจไปที่ลำดับขั้นตอนในการประมวลผลของโปรแกรมเท่านั้น

3. การเขียนโปรแกรมด้วยภาษาคอมพิวเตอร์

การเขียนโปรแกรมเป็นการนำเอาผลลัพธ์ของการออกแบบโปรแกรมมาเปลี่ยนเป็นโปรแกรมภาษาคอมพิวเตอร์ภาษาใดภาษาหนึ่ง ผู้เขียนโปรแกรมจะต้องให้ความสนใจต่อรูปแบบคำสั่งและกฎเกณฑ์ของภาษาที่ใช้ เพื่อให้การประมวลผลเป็นไปตามผลลัพธ์ที่ได้ออกแบบไว้ นอกจากนี้ผู้เขียนโปรแกรมควรแทรกคำอธิบายการทำงานต่างๆ ลงในโปรแกรมเพื่อให้โปรแกรมนั้นมีความกระจ่างชัดและง่ายต่อการตรวจสอบและ โปรแกรมนี้ยังใช้เป็นส่วนหนึ่งของเอกสารประกอบ

4. การทดสอบและแก้ไขโปรแกรม

การทดสอบโปรแกรมเป็นการนำโปรแกรมที่ลงรหัสแล้วเข้า คอมพิวเตอร์ เพื่อตรวจสอบรูปแบบกฎเกณฑ์ของภาษา และผลการทำงานของโปรแกรมนั้น ถ้าพบว่ายังไม่ถูกต้องก็แก้ไขให้ถูกต้องต่อไป ขั้นตอนการทดสอบและแก้ไขโปรแกรม อาจแบ่งได้เป็น 3 ชั้น

1. สร้างแฟ้มเก็บโปรแกรมซึ่งส่วนใหญ่นิยมนำโปรแกรมเข้าผ่านทางแป้นพิมพ์โดยใช้โปรแกรมประมวลคำ
2. ใช้ตัวแปลภาษาคอมพิวเตอร์แปลโปรแกรมที่สร้างขึ้นเป็นภาษาเครื่อง โดยระหว่างการแปลจะมีการตรวจสอบความถูกต้องของรูปแบบและกฎเกณฑ์ในการใช้ภาษา ถ้าคำสั่งใดมีรูปแบบไม่ถูกต้องก็จะแสดงข้อผิดพลาดออกมาเพื่อให้ผู้เขียนนำไป แก้ไขต่อไป ถ้าไม่มีข้อผิดพลาด เราจะได้โปรแกรมภาษาเครื่องที่สามารถให้คอมพิวเตอร์ประมวลผลได้
3. ตรวจสอบความถูกต้องของการประมวลผลของโปรแกรม โปรแกรมที่ถูกต้องตามรูปแบบและกฎเกณฑ์ของภาษา แต่อาจให้ผลลัพธ์ของการประมวลผลไม่ถูกต้องก็ได้ ดังนั้นผู้เขียนโปรแกรมจำเป็นต้องตรวจสอบว่าโปรแกรมประมวลผลถูกต้องตามต้องการหรือไม่ วิธีการหนึ่งก็คือ สมมติข้อมูลตัวแทนจากข้อมูลจริงนำไปให้โปรแกรมประมวลผลแล้วตรวจสอบผลลัพธ์ ว่าถูกต้องหรือไม่ ถ้าพบว่ามีข้อผิดพลาดก็ต้องดำเนินการแก้ไขโปรแกรมต่อไป การสมมติข้อมูลตัวแทนเพื่อการทดสอบเป็นสิ่งที่มีความสำคัญเป็นอย่างมาก ลักษณะของข้อมูลตัวแทนที่ดีควรจะสมมติทั้งข้อมูลที่ถูกต้องและข้อมูลที่ผิดพลาด เพื่อทดสอบว่าโปรแกรมที่พัฒนาขึ้นสามารถครอบคลุมการปฏิบัติงานในเรื่องนั้นๆ ได้ครบถ้วน นอกจากนี้อาจตรวจสอบการทำงานของโปรแกรมด้วยการสมมติตัวเองเป็นคอมพิวเตอร์ที่จะประมวลผล แล้วทำตามคำสั่งทีละคำสั่งของโปรแกรมนั้นๆ วิธีการนี้อาจทำได้ยากถ้าโปรแกรมมีขนาดใหญ่ หรือมีการประมวลผลที่ซับซ้อน

5. การทำเอกสารประกอบโปรแกรม

การทำเอกสารประกอบโปรแกรมเป็นงานที่สำคัญของการพัฒนาโปรแกรม เอกสารประกอบโปรแกรมช่วยให้ผู้ใช้โปรแกรมเข้าใจวัตถุประสงค์ ข้อมูลที่จะต้องใช้กับโปรแกรม ตลอดจนผลลัพธ์ที่จะได้จากโปรแกรม การทำโปรแกรมทุกโปรแกรมจึงควรต้องทำเอกสารกำกับ เพื่อใช้สำหรับการอ้างอิง เมื่อจะใช้งานโปรแกรมและเมื่อต้องการแก้ไขปรับปรุงโปรแกรม เอกสารประกอบโปรแกรมที่จัดทำควรประกอบด้วยหัวข้อต่อไปนี้

1. วัตถุประสงค์
2. ประเภทและชนิดของคอมพิวเตอร์และอุปกรณ์ที่ใช้ในโปรแกรม
3. วิธีการใช้โปรแกรม
4. แนวคิดเกี่ยวกับการออกแบบโปรแกรม
5. รายละเอียดโปรแกรม
6. ข้อมูลตัวแทนที่ใช้ทดสอบ
7. ผลลัพธ์ของการทดสอบ

6. การบำรุงรักษาโปรแกรม

เมื่อโปรแกรมผ่านการตรวจสอบตามขั้นตอนเรียบร้อยแล้ว และถูกนำมาให้ผู้ใช้ได้ใช้งาน ในช่วงแรกผู้ใช้อาจจะยังไม่คุ้นเคยก็อาจทำให้เกิดปัญหาขึ้นมาบ้าง ดังนั้นจึงต้องมีผู้คอยควบคุมดูแลและคอยตรวจสอบการทำงาน การบำรุงรักษาโปรแกรมจึงเป็นขั้นตอนที่ผู้เขียนโปรแกรมต้องคอยเฝ้าดูและหาข้อผิดพลาดของโปรแกรมในระหว่างที่ผู้ใช้ใช้งานโปรแกรม และปรับปรุงโปรแกรมเมื่อเกิดข้อผิดพลาดขึ้น หรือในการใช้งานโปรแกรมไปนานๆ ผู้ใช้อาจต้องการเปลี่ยนแปลงการทำงานของระบบงานเดิม เพื่อให้เหมาะสมกับ เหตุการณ์ นักเขียนโปรแกรมก็จะต้องคอยปรับปรุงแก้ไขโปรแกรมตามความต้องการของผู้ใช้ที่เปลี่ยนแปลงไปนั่นเอง

