บทที่ 4

ชุดโปรแกรมทฤษฎีจำนวน

ชุดโปรแกรมทฤษฎีจำนวน คือชุดโปรแกรมที่ช่วยในการคำนวณทางด้านต่างๆ ในวิชาทฤษฎีจำนวนที่ส่วนของการคำนวณเขียนโดยภาษาปาสคาล และส่วนของการติดต่อผู้ใช้สร้างโดยบอร์แลนเดลไฟ(Borland Delphi) โดยทำการซ่อนส่วนของการคำนวณที่ซับซ้อนบางขั้นตอนไว้ และแสดงขั้นตอนการคำนวณบางส่วนแก่ผู้ใช้ เพื่อช่วยในการคำนวณเพื่อหาคำตอบ การเรียนการสอนตลอดจนการประยุกต์ใช้ทฤษฎีจำนวนในการทำงานด้านต่างๆ เช่นการเข้ารหัสและการถอดรหัสข้อมูลต่างๆ การจัดตารางการแข่งขัน เป็นต้น

การทำงานของชุดโปรแกรมทฤษฎีจำนวนทำให้ผู้ใช้สะดวกและใช้งานง่าย ผู้ใช้ไม่จำเป็นต้องมีความรู้ในการเขียนโปรแกรม เพียงแค่ป้อนค่าข้อมูลนำเข้าต่างๆ ที่แต่ละส่วนโปรแกรมย่อยต้องการในการทำงาน คำตอบและขั้นตอนการคำนวณต่างๆก็จะแสดงออกมาให้ผู้ใช้ทราบ

4.1 การรวบรวมและวิเคราะห์ความต้องการของผู้ใช้

จากการรวบรวมและวิเคราะห์ถึงความต้องการของผู้ใช้ ทั้งจากการสอบถาม การทำโปรโตไทป์(Prototype) และจากการศึกษาในเนื้อหาวิชาทฤษฎีจำนวนของผู้พัฒนาเอง สามารถสรุปถึงความต้องการของผู้ใช้ที่มีต่อชุดโปรแกรมทฤษฎีจำนวนได้ดังต่อไปนี้

4.1.1 การหาตัวหารร่วมมากโดยขั้นตอนวิธีของยูคลิด ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มสองจำนวน และต้องการผลลัพธ์เป็นค่าตัวหารร่วมมากของข้อมูลเข้าทั้งสอง พร้อมทั้งแสดงขั้นตอนการทำย้อนกลับ(Backward) ของขั้นตอนวิธีของยูคลิดอีกด้วย

4.1.2 การหาจำนวนเฉพาะตั้งแต่ 2 จนถึงจำนวนที่กำหนด ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มบวกหนึ่งจำนวน และต้องการผลลัพธ์คือการแสดงจำนวนเฉพาะตั้งแต่ 2 จนถึงจำนวนที่รับเข้าไป ในส่วนนี้มีความจำเป็นต้องกำหนดขอบเขตของข้อมูลที่รับเข้าไปด้วย เนื่องจากหากข้อมูลเข้าเป็นจำนวนที่มีค่ามากเกินไปก็จะใช้เวลาในการแสดงผลมาก อีกทั้งในบางครั้งความต้องการของผู้ใช้ก็เพียงต้องการทดสอบว่าค่าข้อมูลที่ป้อนเข้าไปนั้นเป็นจำนวนเฉพาะหรือไม่เท่านั้น ดังนั้นในส่วนที่เกี่ยวข้องกับจำนวนเฉพาะนี้จึงต้องมีการแยกการทำงานเป็นสองส่วนย่อยคือส่วนการแสดงจำนวน

เฉพาะที่น้อยกว่าหรือเท่ากับจำนวนที่ผู้ใช้ป้อนเข้าไป และส่วนการทดสอบว่าจำนวนที่ผู้ใช้ป้อนเข้าไปนั้นเป็นจำนวนเฉพาะหรือไม่

4.1.3 การหาตัวหารร่วมมากระหว่างจำนวนตั้งแต่ 2 จำนวนขึ้นไป ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มสองจำนวน และต้องการผลลัพธ์เป็นค่าตัวหารร่วมมากของข้อมูลเข้าทั้งสอง โดยไม่สนใจวิธีการว่าจะใช้วิธีใดในการหาคำตอบ ต้องการเพียงผลการคำนวณเท่านั้น

4.1.4 การหาค่าฟังก์ชันฟี(ϕ) เทา(τ)และซิกม่า(σ) ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มบวกหนึ่งจำนวน และต้องการผลลัพธ์เป็นคำตอบที่เกิดจากการคำนวณของฟังก์ชันทั้งสามโดยมีค่าข้อมูลที่ป้อนเข้าไปเป็นค่าพารามิเตอร์ของฟังก์ชัน

4.1.5 การแยกตัวประกอบโดยวิธีของโพลาร์ด (Polard Rho Factorization) ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มบวกหนึ่งจำนวน และต้องการผลลัพธ์เป็นตัวเลขซึ่งคือตัวประกอบตัวหนึ่งของข้อมูลที่ผู้ใช้ป้อนเข้าไป และแสดงออกมาในรูปของผลคูณของจำนวนสองจำนวนที่เป็นตัวประกอบของจำนวนที่ป้อนเข้าไป

4.1.6 ระบบคอนกรูเอนซ์ (Congruence) ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มสามจำนวน คือค่าของตัวเลข A, B และ M ที่สอดคล้องกับสมภาค $AX \equiv B \pmod{M}$ และต้องการผลลัพธ์เป็นค่าของตัวแปร X ที่เกิดจากการคำนวณโดยใช้ค่าข้อมูลเข้าทั้งสาม นอกจากนั้นในส่วนของระบบคอนกรูเอนซ์นี้ ผู้ใช้ยังมีความต้องการในส่วนการทดสอบว่าจำนวนเต็มสองจำนวนที่ผู้ใช้ป้อนให้กับชุดโปรแกรมจะสมภาคกันภายใต้มอดูโลที่กำหนดหรือไม่ ดังนั้นจึงต้องแยกการคำนวณออกมาเป็นอีกส่วนหนึ่งที่ยังคงรับข้อมูลเข้าเป็นเลขจำนวนเต็มสามจำนวนคือ ค่าของตัวเลข A, B และ M เพื่อจะทดสอบว่า A จะคอนกรูเอนซ์กับ B ภายใต้มอดูโล M ที่กำหนดหรือไม่

4.1.7 ขั้นตอนวิธีเศษเหลือน้อยที่สุด(Least Remainder Algorithm) หัวข้อนี้เป็นการหาค่าตัวหารร่วมมากของจำนวนสองจำนวนวิธีหนึ่งที่คล้ายคลึงกับขั้นตอนวิธีของยูคลิด แต่มีรายละเอียดที่แตกต่างกันบ้างในส่วนของการคำนวณ ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มสองจำนวน และ

ต้องการผลลัพธ์เป็นค่าตัวหารร่วมมากของข้อมูลเข้าทั้งสอง พร้อมทั้งแสดงขั้นตอนการทำย้อนกลับ(Backward) ของขั้นตอนวิธีเศษเหลือน้อยที่สุดอีกด้วย

4.1.8 สมการไดโอแฟนไทน์ (Diophantine) ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มสามจำนวนคือ A, B และC ที่ต้องการตามสมการ AX + BY =C และต้องการผลลัพธ์เป็นค่าของ X และ Y ที่เป็นผลจากการคำนวณ นอกจากนั้นยังต้องการแสดงผลการพิจารณาว่าค่าข้อมูลรับเข้ามานั้นสามารถหาคำตอบของสมการนี้ได้หรือไม่ โดยการพิจารณาจากค่าตัวหารร่วมมากของ A และ B ว่าสามารถหาร C ได้ลงตัวหรือไม่ หากสามารถหารได้ลงตัวจึงค่อยคำนวณหาผลเฉลยของสมการ หากไม่สามารถหาผลเฉลยได้ก็ให้แจ้งให้ผู้ใช้ทราบ

4.1.9 การเปลี่ยนเลขฐาน ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มสองจำนวน คือค่าของตัวเลขฐานสิบที่ต้องการจะเปลี่ยนเลขฐานและค่าของฐานที่ต้องการให้เปลี่ยนซึ่งเป็นเลขระหว่าง 2 ถึง 10 และต้องการผลลัพธ์เป็นค่าตัวเลขที่เปลี่ยนเป็นฐานที่ต้องการแล้ว

4.1.10 Round Robin Tournament ในหัวข้อนี้เป็นการประยุกต์ใช้ระบบคอนกรูเอนซ์มาช่วยในการจัดตารางการแข่งขันแบบพบกันหมด ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มบวกหนึ่งจำนวนซึ่งคือจำนวนทีมที่เข้าแข่งขันและต้องการจะจัดตารางการแข่งขัน และต้องการผลลัพธ์เป็นลักษณะตารางการแข่งขันแสดงรอบการแข่งขันแต่ละรอบว่าแต่ละทีมต้องแข่งขันกับทีมใดในรอบการแข่งขันนั้นๆ

4.1.11 การเปลี่ยนจำนวนตรรกยะเป็นเศษส่วนต่อเนื่องจำกัด ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มสองจำนวน คือ A และB ซึ่งสอดคล้องกับเศษส่วน $\frac{A}{B}$ และต้องการผลลัพธ์เป็นเศษส่วนต่อเนื่องซึ่งเขียนอยู่ในรูปแบบเศษส่วนต่อเนื่องจำกัดอย่างง่าย

4.1.12 การเปลี่ยนเศษส่วนต่อเนื่องจำกัดเป็นจำนวนตรรกยะ ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือค่าของพจน์ต่างๆ ของเศษส่วนต่อเนื่อง ซึ่งอยู่ในรูปแบบเศษส่วนต่อเนื่องจำกัดอย่างง่าย จำนวนพจน์ที่รับเข้ามีไม่เกิน 20 พจน์ และต้องการผลลัพธ์การคำนวณเป็นจำนวนตรรกยะซึ่งอยู่ในรูปเศษส่วนทั่วไป

4.1.13 การเปลี่ยนเศษส่วนต่อเนื่องอนันต์เป็นจำนวนอตรรกยะ ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือค่าของพจน์ต่างๆ ของเศษส่วนต่อเนื่อง ซึ่งอยู่ในรูปแบบเศษส่วนต่อเนื่องอนันต์ จำนวนพจน์ที่รับเข้ามีไม่เกิน 20 พจน์ และต้องการผลลัพธ์การคำนวณเป็นจำนวนอตรรกยะซึ่งอยู่ในรูปจำนวนจริงมีทศนิยมตามหลัง

4.1.14 การเปลี่ยนจำนวนอตรรกยะเป็นเศษส่วนต่อเนื่องอนันต์ ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มสามจำนวนคือ A, B และC ที่ต้องการซึ่งสอดคล้องกับรูปแบบของจำนวนอตรรกยะ $\frac{A+\sqrt{B}}{C}$ และต้องการผลลัพธ์เป็นค่าของเศษส่วนต่อเนื่องอนันต์ที่เป็นผลจากการคำนวณ

4.1.15 การแยกตัวประกอบของจำนวนต่างๆ ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มบวกหนึ่งจำนวน และต้องการผลลัพธ์เป็นผลคูณระหว่างจำนวนเฉพาะซึ่งคือตัวประกอบของข้อมูลที่ผู้ใช้ป้อนเข้าไป

4.1.16 การแยกตัวประกอบโดยวิธีของแฟร์มาต์ (Fermat Factorization) ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มบวกหนึ่งจำนวน และต้องการผลลัพธ์เป็นผลคูณระหว่างจำนวนสองจำนวนซึ่งคือตัวประกอบของข้อมูลที่ผู้ใช้ป้อนเข้าไป ทั้งยังต้องการให้แสดงขั้นตอนวิธีในการคำนวณแต่ละขั้นตอนตามวิธีของแฟร์มาต์อีกด้วย

4.1.17 การแยกตัวประกอบโดยการใช้เศษส่วนต่อเนื่อง ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือตัวเลขจำนวนเต็มบวกหนึ่งจำนวน และต้องการผลลัพธ์เป็นผลคูณระหว่างจำนวนสองจำนวนซึ่งคือตัวประกอบของข้อมูลที่ผู้ใช้ป้อนเข้าไป ทั้งยังต้องการให้แสดงขั้นตอนวิธีในการคำนวณแต่ละขั้นตอนตามวิธีของการแยกตัวประกอบโดยใช้เศษส่วนต่อเนื่องอีกด้วย

4.1.18 การเข้ารหัสแบบซีซ่า (Caesar Encryption) ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือ ค่าคีย์ที่ใช้ในการเข้ารหัส และข้อความที่ต้องการเข้ารหัส และต้องการผลลัพธ์การทำงานเป็นข้อความที่ผ่านการเข้ารหัสเรียบร้อยแล้ว และในทางกลับกัน คือการถอดรหัส ผู้ใช้ต้องการป้อนข้อมูลเข้าคือค่าคีย์ที่ใช้ในการถอดรหัส และข้อความที่ต้องการถอดรหัส โดยต้องการผลลัพธ์การทำงานเป็นข้อความที่ผ่านการถอดรหัสเรียบร้อยแล้ว

4.1.19 การเข้ารหัสแบบบล็อค (Block Encryption) ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือ ค่าคีย์ที่ใช้ในการเข้ารหัสซึ่งเป็นเมตริกซ์(Matrix) ขนาด 2x2 และข้อความที่ต้องการเข้ารหัส และต้องการผลลัพธ์การทำงานเป็นข้อความที่ผ่านการเข้ารหัสเรียบร้อยแล้ว และในทางกลับกัน คือการถอดรหัส ผู้ใช้ต้องการป้อนข้อมูลเข้าคือค่าคีย์ที่ใช้ในการถอดรหัสเป็นเมตริกซ์(Matrix) ขนาด 2x2 เช่นเดียวกัน และข้อความที่ต้องการถอดรหัส โดยต้องการผลลัพธ์การทำงานเป็นข้อความที่ผ่านการถอดรหัสเรียบร้อยแล้ว

4.1.20 การเข้ารหัสแบบแอฟไฟน์ (Affine Encryption) ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือ ค่าคีย์ที่ใช้ในการเข้ารหัสซึ่งเป็นค่าตัวเลขจำนวนเต็มสองจำนวน และข้อความที่ต้องการเข้ารหัส และต้องการผลลัพธ์การทำงานเป็นข้อความที่ผ่านการเข้ารหัสเรียบร้อยแล้ว และในทางกลับกัน คือการถอดรหัส ผู้ใช้ต้องการป้อนข้อมูลเข้าคือค่าคีย์ที่ใช้ในการถอดรหัสเป็นค่าตัวเลขจำนวนเต็มเช่นเดียวกัน และข้อความที่ต้องการถอดรหัส โดยต้องการผลลัพธ์การทำงานเป็นข้อความที่ผ่านการถอดรหัสเรียบร้อยแล้ว

4.1.21 การเข้ารหัสแบบแนบแสค(Knapsack Encryption) ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือ ค่าต่างๆ ที่จำเป็นในการเข้ารหัสซึ่งคือลำดับซูเปอร์อินครีสซิง(Super Increasing) ค่าของตัว M ซึ่งเป็นจำนวนเต็มที่มากกว่า 58 และข้อความที่ต้องการเข้ารหัส และต้องการผลลัพธ์การทำงานเป็นข้อความที่ผ่านการเข้ารหัสเรียบร้อยแล้ว และในทางกลับกัน คือการถอดรหัส ผู้ใช้ต้องการป้อนข้อมูลเข้าคือค่าต่างๆ ที่ใช้ในการถอดรหัสเป็นค่าตัวเลขจำนวนเต็ม และข้อความที่ต้องการถอดรหัส โดยต้องการผลลัพธ์การทำงานเป็นข้อความที่ผ่านการถอดรหัสเรียบร้อยแล้ว

4.1.22 การเข้ารหัสแบบอาร์เอสเอ (RSA Encryption) ในส่วนหัวข้อนี้ผู้ใช้ต้องการป้อนค่าข้อมูลเข้าคือ ค่าคีย์ที่ใช้ในการเข้ารหัสเป็นค่าตัวเลขจำนวนเต็มบวกค่าจำนวนเฉพาะสองจำนวนที่ใช้ในการทำงาน และข้อความที่ต้องการเข้ารหัส และต้องการผลลัพธ์การทำงานเป็นข้อความที่ผ่านการเข้ารหัสเรียบร้อยแล้ว และในทางกลับกัน คือการถอดรหัส ผู้ใช้ต้องการป้อนข้อมูลเข้าคือค่าคีย์ที่ใช้ในการถอดรหัสเป็นค่าตัวเลขจำนวนเต็ม และข้อความที่ต้องการถอดรหัส โดยต้องการผลลัพธ์การทำงานเป็นข้อความที่ผ่านการถอดรหัสเรียบร้อยแล้ว

4.1.23 Elliptic Curve และ Quadratic Residue ในส่วนหัวข้อนี้สามารถแบ่งได้เป็นสองส่วนย่อยคือ ส่วนของ Elliptic Curve และส่วนของ Quadratic Residue โดยที่ส่วนของ Quadratic Residue จะช่วยหาคำตอบในบางส่วนของ Elliptic Curve แต่ในบางโอกาสผู้ใช้อาจต้องการจะใช้งานเฉพาะส่วนของ Quadratic Residue เท่านั้นจึงต้องแยกส่วนของ Quadratic Residue เอาไว้เป็นพิเศษอีกส่วนหนึ่ง ในส่วนของ Quadratic Residue ผู้ใช้ต้องการป้อนข้อมูลเข้าคือ ค่าตัวเลขจำนวนเต็มหนึ่งจำนวนคือ A และจำนวนเฉพาะหนึ่งจำนวนคือ P ที่สอดคล้องกับสมภาค $X^2 \equiv A \pmod{P}$ และผลลัพธ์การทำงานที่ต้องการคือค่าของตัวแปร X และในส่วนของ Elliptic Curve ผู้ใช้ต้องการป้อนข้อมูลเข้าคือ ค่าตัวเลขจำนวนเต็มสองจำนวนคือ A และ B และจำนวนเฉพาะอีกหนึ่งจำนวนคือ P ที่สอดคล้องกับสมภาค $Y^2 \equiv X^3 + AX + B \pmod{P}$ ในส่วนของการแสดงผลผู้ใช้ต้องการให้แสดงผลเฉลยของสมภาคนี้ว่ามีผลเฉลยหรือไม่ และหากมีผลเฉลยก็ให้แสดงออกมา พร้อมทั้งแสดงคู่อันดับทั้งหมดที่เป็นพิกัดจุดบน Elliptic Curve จากนั้นก็ให้นำคู่อันดับทั้งหมดนั้นมาทดสอบว่ามีคู่อันดับใดบ้างที่เป็นเจเนอเรเตอร์ (Generator)

4.2 การออกแบบชุดโปรแกรม

จากการรวบรวมและวิเคราะห์ถึงความต้องการของผู้ใช้ สามารถสรุปการทำงานของชุดโปรแกรมได้เป็นกลุ่มๆ คือ

- กลุ่มที่ทำงานเกี่ยวกับการแยกตัวประกอบ
  - Polard Rho Factorization
  - Fermat Factorization
  - Prime Factorization
  - Continued Fraction Factorization
- กลุ่มที่ทำงานเกี่ยวกับวิชาทฤษฎีจำนวนทั่วไป
  - การหาค่าของฟังก์ชันพิเศษต่างๆ
  - ระบบคอนกรูเอนซ์
  - การหาค่าตัวหารร่วมมาก
- กลุ่มการประยุกต์ใช้วิชาทฤษฎีจำนวน

- Round Robin Tournament
- การเปลี่ยนเลขฐาน

● กลุ่มการทำงานเกี่ยวกับระบบรหัสลับ

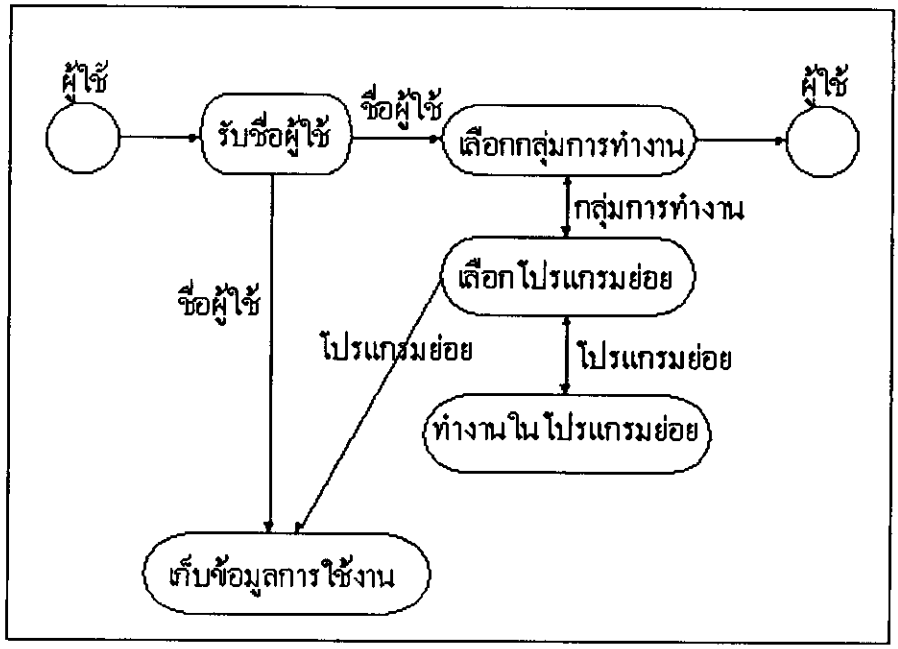
- Caesar Encryption
- Block Encryption
- Affine Encryption
- Knapsack Encryption
- RSA Encryption

● กลุ่มการทำงานเกี่ยวกับเศษส่วนต่อเนื่อง

- การเปลี่ยนจำนวนตรรกยะเป็นเศษส่วนต่อเนื่อง
- การเปลี่ยนเศษส่วนต่อเนื่องเป็นจำนวนตรรกยะ
- การเปลี่ยนเศษส่วนต่อเนื่องเป็นจำนวนอตรรกยะ
- การเปลี่ยนจำนวนอตรรกยะเป็นเศษส่วนต่อเนื่อง

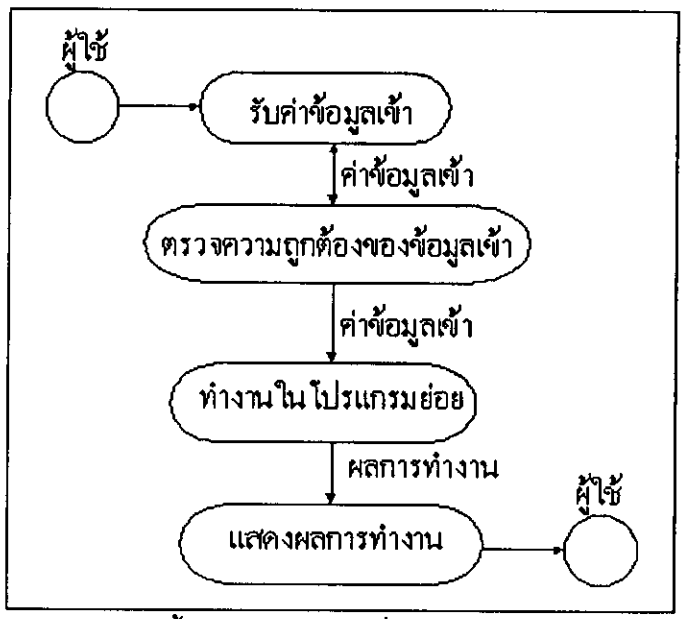
● Elliptic Curve and Quadratic Residue

ชุดโปรแกรมจึงมีรูปแบบเป็นการแบ่งกลุ่มการทำงาน เมื่อเลือกกลุ่มการทำงานใดๆ แล้วจึงแสดงส่วนการทำงานย่อยที่จะใช้ในการคำนวณในหัวข้อต่างๆ ของสาขาวิชาทฤษฎีจำนวน รูปแบบการทำงานเกี่ยวกับชุดโปรแกรมดังแสดงตามแผนภาพที่ 4.1



แผนภาพ 4.1 แสดงการทำงานโดยรวมของชุดโปรแกรม

ในส่วนของโปรแกรมย่อยต่างๆ ที่ใช้ทำงานในการคำนวณหัวข้อต่างๆ ของวิชาทฤษฎี จำนวนนั้นจะประกอบด้วยขั้นตอนต่างๆ โดยทั่วไปดังแสดงในแผนภาพที่ 4.2

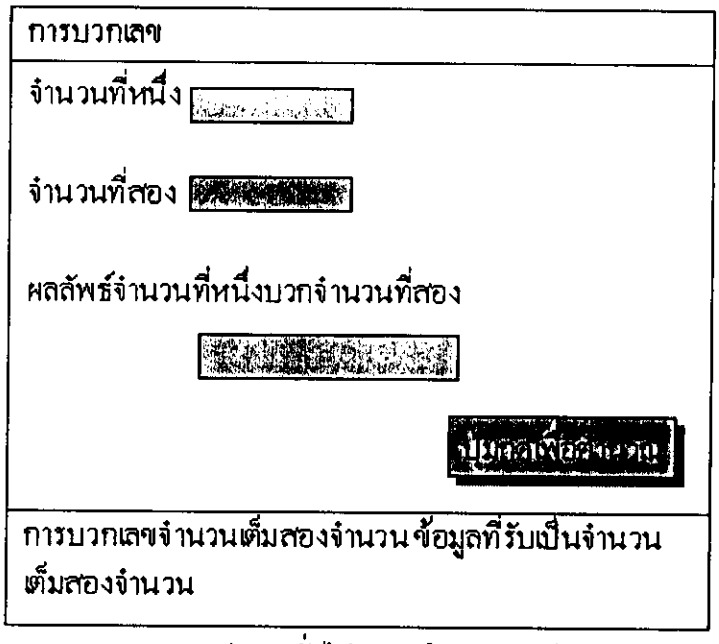


แผนภาพ 4.2 แสดงขั้นตอนต่างๆ โดยทั่วไปของโปรแกรมย่อยต่างๆ

4.3 การออกแบบส่วนรับข้อมูลเข้าและส่วนแสดงผลการทำงาน

ในส่วนของการรับข้อมูลเข้าและส่วนแสดงผลการทำงานข้อมูลต่างๆ ของชุดโปรแกรมทฤษฎีจำนวนนั้นผู้พัฒนาได้เลือกพัฒนาเป็นลักษณะหน้าต่าง (Windows) ที่แสดงข้อมูลส่วนการ

คำนวณต่างๆ ขึ้นมา ที่ส่วนหัวของหน้าต่างจะมีการบอกชื่อว่าเป็นหน้าต่างที่ทำงานในหัวข้อการคำนวณอะไร ทุกๆ หน้าต่างที่ปรากฏขึ้นมานั้นจะมีทั้งส่วนที่เป็นการรับข้อมูลเข้าและแสดงผลการทำงานแสดงอยู่ในหน้าต่างนั้นเลย การจะเปลี่ยนการทำงานไปยังส่วนโปรแกรมย่อยที่ใช้ทำงานด้านอื่นๆ จะต้องปิดหน้าต่างก่อนหน้าที่ทำงานเสียก่อน นอกจากนั้นในหน้าต่างเดียวกันนั้นยังมีส่วนข้อมูลที่ช่วยในการทำงานในหน้าต่างนั้นๆ ด้วย เช่น ส่วนการช่วยเหลือโดยแนะนำผู้ใช้ว่าควรป้อนข้อมูลประเภทใด และหากยังมีข้อผิดพลาดในการป้อนข้อมูลอีกก็จะแสดงหน้าต่างที่แจ้งข้อผิดพลาดให้ผู้ใช้ทราบอีกครั้ง รูปแบบทั่วไปของหน้าต่างของส่วนโปรแกรมย่อยที่คำนวณในหัวข้อต่างๆ ดังแสดงตามแผนภาพที่ 4.3



แผนภาพ 4.3 รูปแบบทั่วไปของหน้าต่างส่วนโปรแกรมย่อย

ในบางส่วนของโปรแกรมย่อยที่มีการทำงานซ้ำๆ กันหลายครั้งจนได้คำตอบ เช่น การแยกตัวประกอบโดยวิธีของแฟร์มาต์ การแยกตัวประกอบโดยการใช้เศษส่วนต่อเนื่อง ในส่วนการทำงานนี้จะมีเงื่อนไขบางข้อของข้อมูลที่รับเข้าที่เป็นตัวบ่งชี้การทำงานของโปรแกรมย่อย ข้อมูลเข้าบางค่าแม้จะมีค่าที่ถูกต้องตามเงื่อนไข แต่อาจทำให้โปรแกรมย่อยเสียเวลาในการคำนวณมากเนื่องจากการทำซ้ำเพื่อหาคำตอบของข้อมูลเข้านั้นๆ ดังนั้นในการคำนวณในหัวข้อดังกล่าวจะมีการกำหนดจำนวนการทำซ้ำไว้ หากครบจำนวนรอบการทำงานตามที่กำหนดไว้แล้วยังไม่ได้คำตอบ ก็ให้หยุดการทำงานและแจ้งแก่ผู้ใช้เพื่อไม่ให้เสียเวลาในการคำนวณมากนัก

4.4 **การทำโพรโตไทป์** (Prototype)

ภายหลังจากได้ทำการรวบรวมและวิเคราะห์ความต้องการของผู้ใช้ และหลังจากได้ทำการวางแผนและออกแบบชุดโปรแกรมแล้ว ผู้พัฒนาจึงเห็นว่าเพื่อให้ผู้ใช้ได้เห็นแนวทางหรือเกิดภาพว่าชุดโปรแกรมทำงานอย่างไร การรับข้อมูลเข้าและแสดงผลข้อมูลตรงตามที่ผู้ใช้ต้องการหรือไม่ และแสดงการทำงานต่างๆ ของชุดโปรแกรมได้ชัดเจนยิ่งขึ้น

ระยะที่ 1 แยกชุดโปรแกรมโดยรวมให้เป็นโปรแกรมย่อย เพื่อไม่ให้เป็นการยากในการทำโพรโตไทป์สำหรับชุดโปรแกรมขึ้นมาในครั้งเดียว ดังนั้นจึงจำเป็นต้องแยกส่วนของชุดโปรแกรมที่เห็นว่าจำเป็นที่ต้องทำเป็นส่วนๆ เพื่อให้ง่ายในการจัดการและวิเคราะห์

ระยะที่ 2 สร้างโพรโตไทป์ เพื่อให้ผู้ใช้ได้เห็นแนวทาง ลักษณะและขั้นตอนการทำงานของชุดโปรแกรมว่าเป็นอย่างไร โดยโพรโตไทป์จะต้องมีลักษณะที่สามารถแก้ไขและเปลี่ยนแปลงได้ง่ายตามความเหมาะสม โดยการแก้ไขที่จะเกิดขึ้นจะต้องทำให้ชุดโปรแกรมเข้าใกล้ความต้องการของผู้ใช้มากที่สุด โดยให้โอกาสผู้ใช้ชุดโปรแกรมร่วมแสดงความคิดเห็นด้วย

ระยะที่ 3 ทดลองใช้โพรโตไทป์ โดยการให้ผู้ใช้ชุดโปรแกรมทดลองใช้โพรโตไทป์ด้วยตนเอง โดยได้มีการศึกษาถึงการตอบโต้และปฏิกิริยาระหว่างผู้ใช้และโพรโตไทป์ เพื่อจะได้เรียนรู้ถึงข้อคิดเห็นที่จะเปลี่ยนแปลงหรือขยายส่วนของโพรโตไทป์ออกไป โดยที่โพรโตไทป์สามารถเปลี่ยนแปลงได้ตลอดเวลา และการเปลี่ยนแปลงโพรโตไทป์นี้เป็นผลมาจากคำแนะนำและปฏิกิริยาตอบสนองของผู้ใช้ระบบ จึงสามารถเชื่อมั่นได้ว่าขั้นตอนต่างๆ ของชุดโปรแกรม เช่น การรับข้อมูลเข้า การคำนวณ วิธีการคำนวณ และการแสดงผลลัพธ์ย่อมต้องได้รับการปรับปรุงให้ตอบสนองกับความต้องการที่แท้จริงของผู้ใช้ และสามารถค้นหาข้อผิดพลาดที่อาจเกิดจากการทำงานของชุดโปรแกรมได้ง่ายยิ่งขึ้น

ในส่วนของการดำเนินการพัฒนาชุดโปรแกรมทฤษฎีจำนวนนั้น เริ่มแรกด้วยการเขียนคำสั่งงานในเทอร์โบปาสคาล (Turbo Pascal) ก่อน ซึ่งเป็นการเขียนคำสั่งงานที่ใช้ในการคำนวณในหัวข้อต่างๆ จากนั้นจึงใช้บอร์แลนด์เดลไฟ (Borland Delphi) ในการสร้างหน้าต่างส่วนที่ใช้ติดต่อกับผู้ใช้ ในการใช้บอร์แลนด์เดลไฟเพื่อสร้างส่วนติดต่อกับผู้ใช้นั้นมีบางส่วนของคำสั่งงานที่ต้องเปลี่ยนแปลงไปบ้างเนื่องจากไวยกรณ์ (Syntax) ในบางส่วนของบอร์แลนด์เดลไฟและเทอร์โบปาสคาลมีความแตกต่างกัน เช่น

คำสั่งรับข้อมูลเข้า

เทอร์โบปาสคาล Readln(X);

บอร์แลนด์เดลไฟ X := strtoint(EditText1.text);

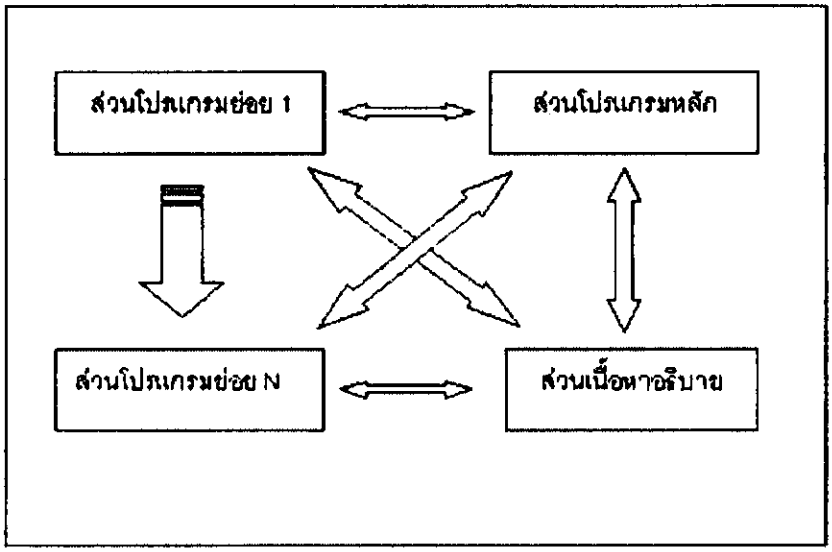
คำสั่งแสดงผลข้อมูล

เทอร์โบปาสคาล Writeln(X);

บอร์แลนด์เดลไฟ Memo1.lines.add(strtoint(EditText1.text));

4.5 การทำงานของชุดโปรแกรมทฤษฎีจำนวน

ลักษณะการทำงานของชุดโปรแกรมทฤษฎีจำนวนแสดงดังภาพประกอบที่ 4.4 เป็นลักษณะการทำงานที่มีส่วนของโปรแกรมหลักที่คอยเรียกใช้การทำงานของโปรแกรมย่อยต่างๆ



ภาพประกอบ 4.4 แสดงโครงสร้างการทำงานของชุดโปรแกรม

4.5.1 ส่วนของโปรแกรมหลัก ประกอบไปด้วย

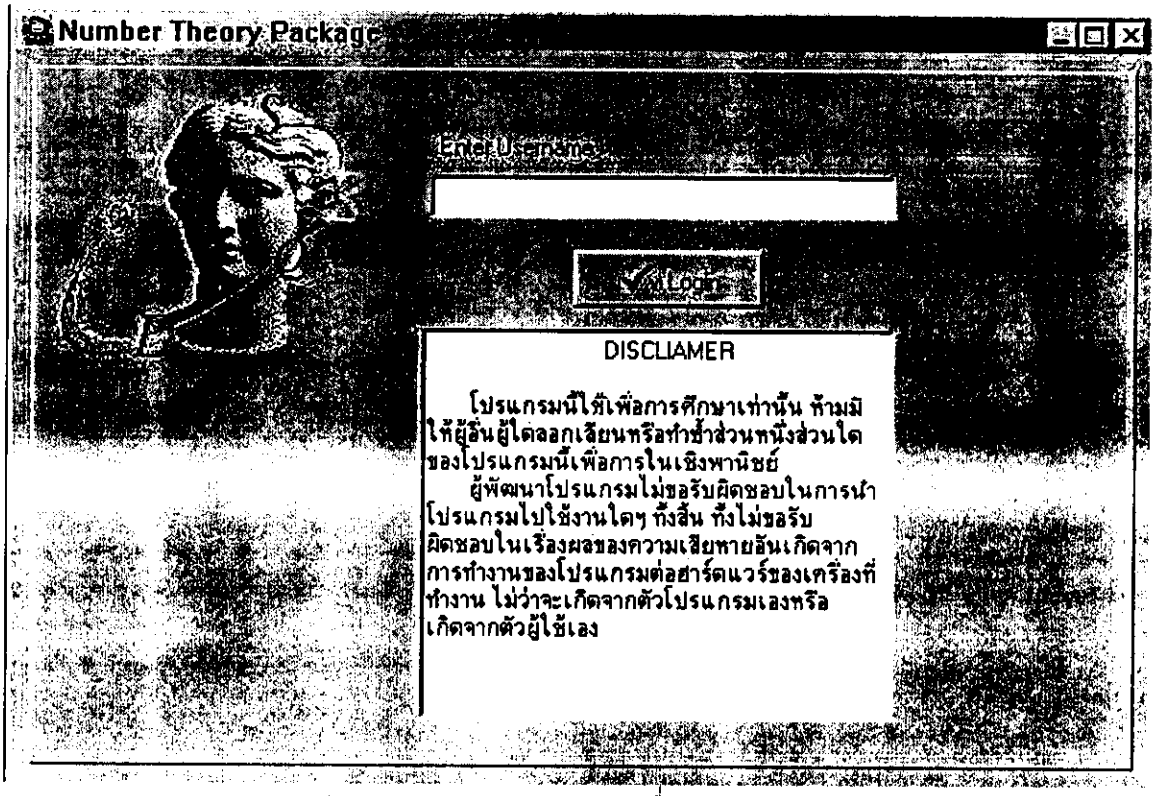
4.5.1.1ส่วนของการลงชื่อเข้าใช้ชุดโปรแกรม เพื่อจะได้ทราบว่ามีผู้ใช้คนใด ได้เข้าใช้ชุดโปรแกรมเมื่อวันที่ใดและเวลาใด ดังภาพประกอบที่ 4.5

4.5.1.2 ส่วนของการเชื่อมโยงไปยังส่วนของเนื้อหาอธิบาย ดังภาพประกอบที่ 4.6

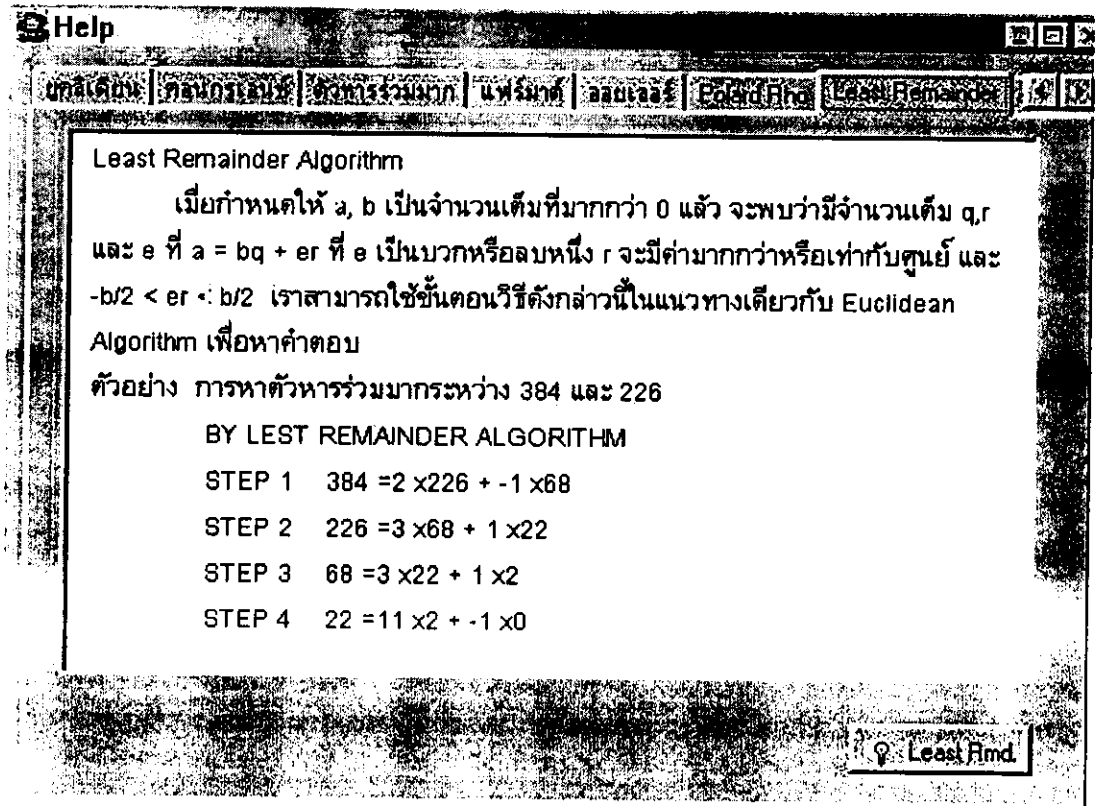
4.5.1.3 ส่วนของการเชื่อมโยงไปยังส่วนการอ้างอิง เนื้อหาในการทำวิจัย ดังภาพประกอบที่ 4.7

4.5.1.4 ส่วนของการเชื่อมโยงไปยังส่วนของโปรแกรมย่อยที่ช่วยในการคำนวณหัวข้อต่างๆ จะรวมกันเป็นกลุ่มการทำงาน ดังภาพประกอบที่ 4.8

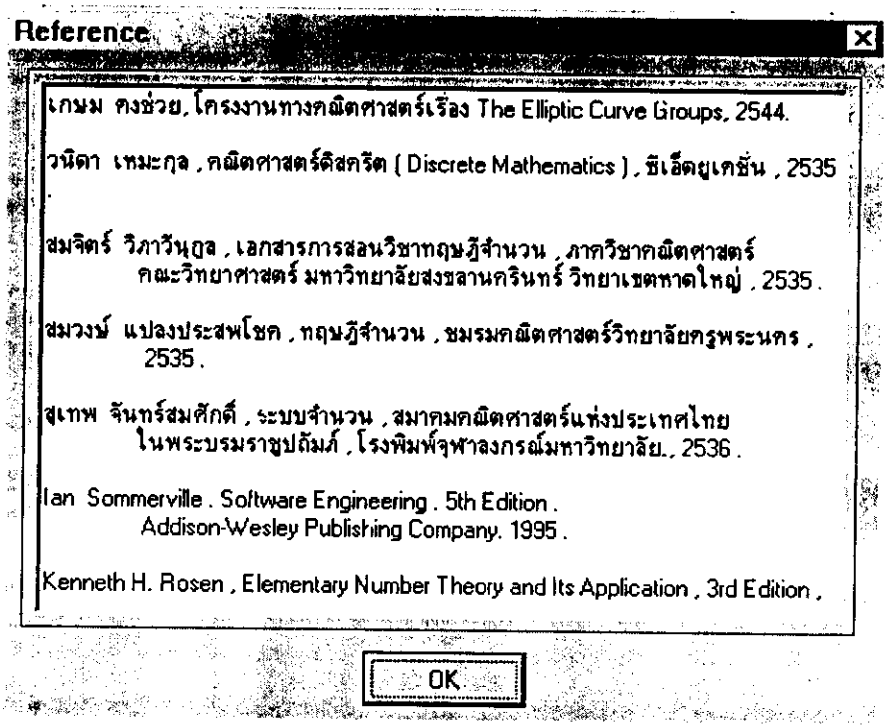
4.5.1.5 ส่วนของการเชื่อมโยงไปยังส่วนที่เกี่ยวกับชุดโปรแกรม ดังภาพประกอบที่ 4.9



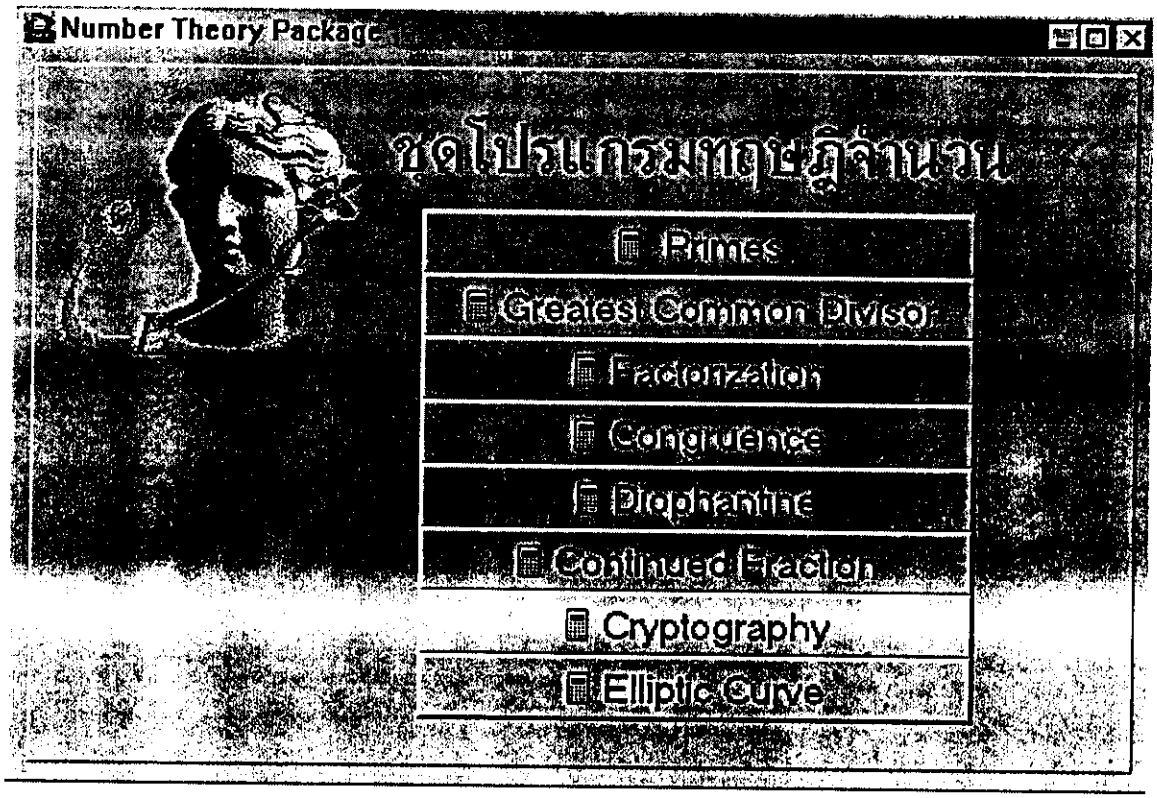
ภาพประกอบ 4.5 ส่วนของการลงชื่อเข้าใช้ชุดโปรแกรม



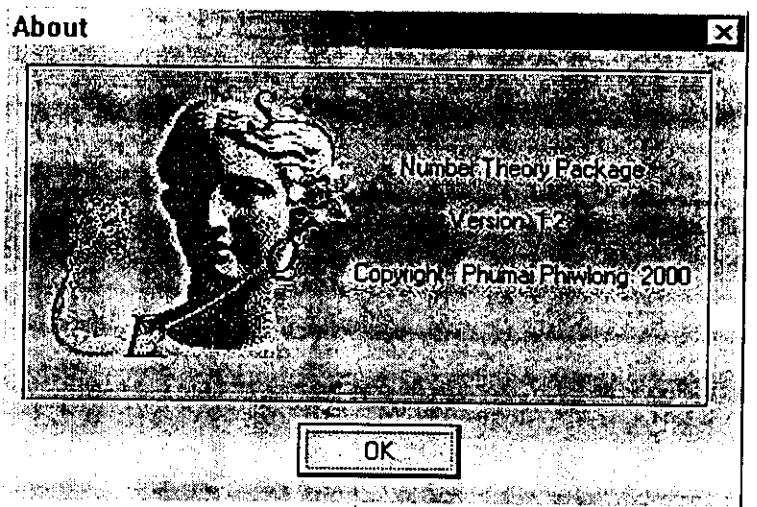
ภาพประกอบ 4.6 ส่วนของการเชื่อมโยงไปยังส่วนเนื้อหาอธิบาย



ภาพประกอบ 4.7 ส่วนของการเชื่อมโยงไปยังส่วนเนื้อหาที่ใช้อ้างอิง



ภาพประกอบ 4.8 ส่วนของการเชื่อมโยงไปยังกลุ่มการทำงานของโปรแกรมย่อย



ภาพประกอบ 4.9 ส่วนของการเชื่อมโยงไปยังส่วนเกี่ยวกับชุดโปรแกรม

4.5.2 ส่วนของโปรแกรมย่อย ประกอบไปด้วยส่วนที่ทำการคำนวณเกี่ยวกับเนื้อหาต่างๆ ในวิชาทฤษฎีจำนวน ดังรายชื่อหัวข้อดังต่อไปนี้

4.5.2.1 การหาตัวหารร่วมมากโดยขั้นตอนวิธีของยูคลิด แฟ้มข้อมูลที่ใช้ทำงานคือ Unit13.pas

4.5.2.2 การหาจำนวนเฉพาะตั้งแต่ 2 จนถึงจำนวนที่กำหนด แฟ้มข้อมูลที่ใช้ทำงานคือ Unit3.pas

4.5.2.3 การหาตัวหารร่วมมากระหว่างจำนวนตั้งแต่ 2 จำนวนขึ้นไป แฟ้มข้อมูลที่ใช้ทำงานคือ Unit4.pas

4.5.2.4 การหาค่าฟังก์ชันฟี(ϕ) เทา(τ)และซิกม่า(σ) แฟ้มข้อมูลที่ใช้ทำงานคือ Unit12.pas

4.5.2.5 การแยกตัวประกอบโดยวิธีของโพลาร์ด (Polard Rho Factorization)แฟ้มข้อมูลที่ใช้ทำงานคือ Unit8.pas

4.5.2.6 ระบบคอนกรูเอนซ์ แฟ้มข้อมูลที่ใช้ทำงานคือ Unit5.pas

4.5.2.7 ขั้นตอนวิธีเศษเหลือน้อยที่สุด(Least Remainder Algorithm) แฟ้มข้อมูลที่ใช้ทำงานคือ Unit15.pas

4.5.2.8 สมการไดโอแฟนไทน์ (Diophantine) แฟ้มข้อมูลที่ใช้ทำงานคือ Unit17.pas

4.5.2.9 การเปลี่ยนเลขฐาน แฟ้มข้อมูลที่ใช้ทำงานคือ Unit20.pas

4.5.2.10 Round Robin Tournament แฟ้มข้อมูลที่ใช้ทำงานคือ Unit25.pas

4.5.2.11 การเปลี่ยนจำนวนตรรกยะเป็นเศษส่วนต่อเนื่องจำกัด แฟ้มข้อมูลที่ใช้ทำงานคือ Unit9.pas

4.5.2.12 การเปลี่ยนเศษส่วนต่อเนื่องจำกัดเป็นจำนวนตรรกยะ แฟ้มข้อมูลที่ใช้ทำงานคือ Unit18.pas

4.5.2.13 การเปลี่ยนเศษส่วนต่อเนื่องอนันต์เป็นจำนวนอตรรกยะ แฟ้มข้อมูลที่ใช้ทำงานคือ Unit19.pas

4.5.2.14 การเปลี่ยนจำนวนอตรรกยะเป็นเศษส่วนต่อเนื่องอนันต์ แฟ้มข้อมูลที่ใช้ทำงานคือ Unit16.pas

4.5.2.15 การแยกตัวประกอบของจำนวนต่างๆ แฟ้มข้อมูลที่ใช้ทำงานคือ Unit6.pas

4.5.2.16 การแยกตัวประกอบโดยวิธีของแฟร์มาต์ แฟ้มข้อมูลที่ใช้ทำงานคือ Unit7.pas

4.5.2.17 การแยกตัวประกอบโดยการใช้เศษส่วนต่อเนื่อง แฟ้มข้อมูลที่ใช้ทำงานคือ Unit21.pas

4.5.2.18 การเข้ารหัสแบบซีซ่า (Caesar Encryption) แฟ้มข้อมูลที่ใช้ทำงานคือ Unit10.pas

4.5.2.19 การเข้ารหัสแบบบล็อค (Block Encryption) แฟ้มข้อมูลที่ใช้ทำงานคือ Unit11.pas

4.5.2.20 การเข้ารหัสแบบแอฟไฟน์ (Affine Encryption) แฟ้มข้อมูลที่ใช้ทำงานคือ Unit27.pas

4.5.2.21 การเข้ารหัสแบบแนบแสค (Knapsack Encryption) แฟ้มข้อมูลที่ใช้ทำงานคือ Unit28.pas

4.5.2.22 การเข้ารหัสแบบอาร์เอสเอ (RSA Encryption) แฟ้มข้อมูลที่ใช้ทำงานคือ Unit29.pas

4.5.2.23 Elliptic Curve และ Quadratic Residue แฟ้มข้อมูลที่ใช้ทำงานคือ Unit31.pas

4.5.3 การเชื่อมโยงการทำงานระหว่างโปรแกรมหลักและโปรแกรมย่อย ในส่วนของการเขียนคำสั่งโปรแกรมจะต้องมีคำสั่งสำหรับการเรียกใช้ยูนิตที่เก็บคำสั่งสำหรับการทำงานของโปรแกรมย่อยต่างๆ ในส่วนหัวของรหัสโปรแกรมหลักดังภาพประกอบที่ 4.10 ซึ่งจะอยู่ในรูปแบบคำสั่ง

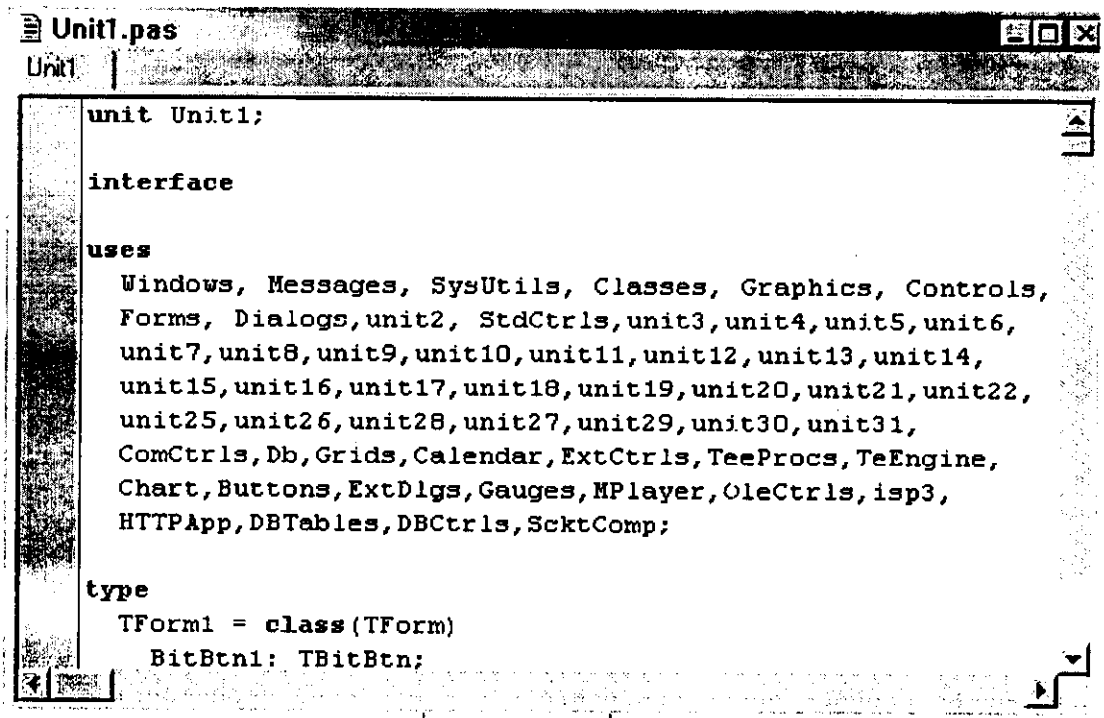
Uses Unit1, Unit2, Unit3, ..., Unit N

เมื่อ N เป็นหมายเลขของยูนิตที่เรียกใช้

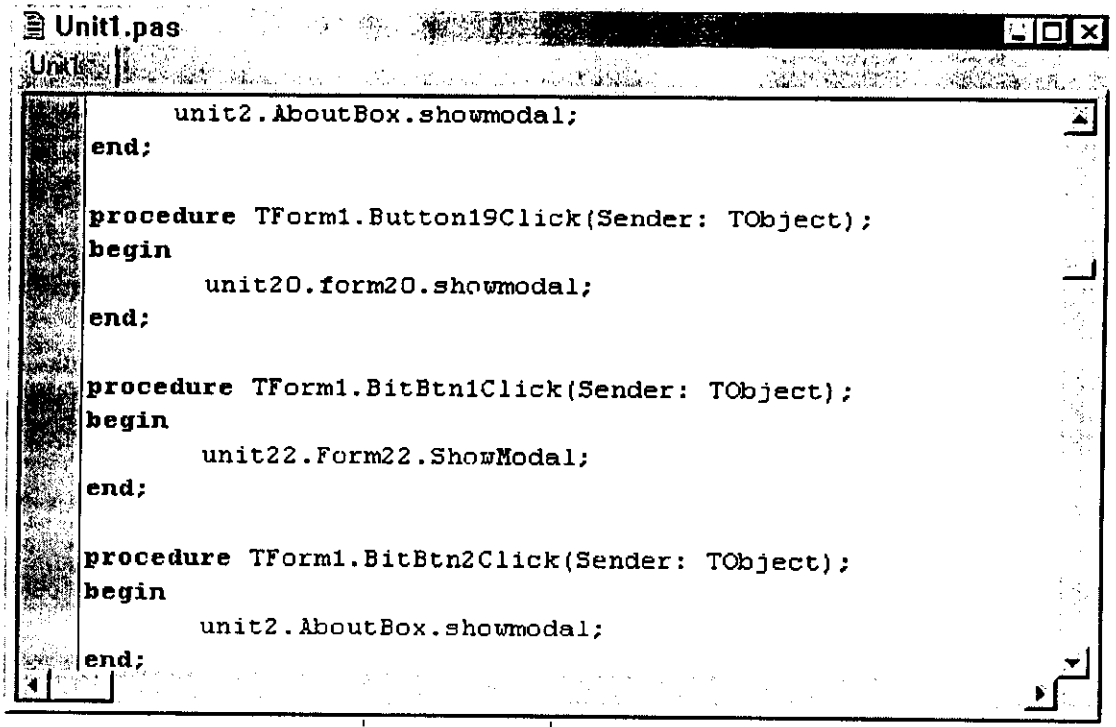
จากนั้นในส่วนการที่จะให้แต่ละโปรแกรมย่อยทำงานนั้นจะมีคำสั่งเรียกใช้อีกในส่วนการทำงานของโปรแกรมหลัก ดังภาพประกอบที่ 4.11 ซึ่งจะอยู่ในรูปแบบคำสั่ง

Unit N . Form N . Showmodal;

เมื่อ N เป็นหมายเลขของยูนิตที่เรียกใช้



ภาพประกอบที่ 4.10 แสดงคำสั่งการเรียกใช้ยูนิตโปรแกรมย่อย



ภาพประกอบที่ 4.11 แสดงคำสั่งการเรียกการทำงานโปรแกรมย่อย

4.6 การทำงานของส่วนโปรแกรมย่อย

โปรแกรมย่อยที่ช่วยในการคำนวณต่างๆ นั้น สามารถแบ่งกลุ่มการทำงานได้ดังนี้

4.6.1 ส่วนที่เป็นการคำนวณหาค่าต่างๆ เช่น การหาค่าตัวหารร่วมมากโดยใช้ขั้นตอนวิธีของยูคลิด การหาค่าตัวหารร่วมมากโดยวิธี Least Remainder Algorithm

4.6.2 ส่วนที่เป็นการคำนวณการแยกตัวประกอบโดยวิธีต่างๆ เช่น การแยกตัวประกอบโดยวิธีของ Polard Rho การแยกตัวประกอบโดยใช้เศษส่วนต่อเนื่อง การแยกตัวประกอบแบบ Prime Factorize

4.6.3 ส่วนที่เป็นการคำนวณเกี่ยวกับเศษส่วนต่อเนื่อง เช่นการแปลงเศษส่วนต่อเนื่องจำกัดเป็นจำนวนตรรกยะและการแปลงจำนวนตรรกยะเป็นเศษส่วนต่อเนื่องจำกัด การแปลงเศษส่วนต่อเนื่องอนันต์เป็นจำนวนอตรรกยะและการแปลงจำนวนอตรรกยะเป็นเศษส่วนต่อเนื่องอนันต์

4.6.4 ส่วนที่เป็นการเข้ารหัสและถอดรหัสแบบต่างๆ เช่น Caesar Encryption, Block cipher Encryption, RSA, Knapsack

4.6.5 ส่วนที่เป็นการคำนวณเกี่ยวกับ Elliptic curve และ Quadratic Residue

4.6.6 ส่วนที่เป็นการประยุกต์ใช้ทฤษฎีจำนวนในงานอื่นๆ เช่น การจัดตารางการแข่งขันแบบ Round Robin Tournament

4.7 ฟังก์ชั่นต่างๆ ที่มีการเรียกใช้งานบ่อย

4.7.1 ฟังก์ชั่นการหาค่าตัวหารร่วมมากระหว่างจำนวนสองจำนวน ฟังก์ชั่นนี้ใช้การทำงานแบบเวียนบังเกิด (Recursive) โดยมีทฤษฎีในการคำนวณอ้างอิงจากการหาตัวหารร่วมมากโดยวิธีของยูคลิด ข้อมูลเข้าเป็นเลขจำนวนเต็มสองจำนวน ผลการทำงานเป็นเลขจำนวนเต็มหนึ่งจำนวนซึ่งคือค่าตัวหารร่วมมากของค่าข้อมูลเข้าทั้งสอง รายละเอียดรหัสโปรแกรมของฟังก์ชั่นเป็นดังต่อไปนี้

```
function gcd(x,y:longint):longint;
begin
        If Y=0 Then GCD := X Else GCD := GCD(Y,X mod Y);
end;
```

4.7.2 ฟังก์ชั่นการหาค่าออยเลอร์ฟี ฟังก์ชั่นนี้เรียกใช้การทำงานการหาค่าตัวหารร่วมมากระหว่างจำนวนสองจำนวนด้วย ข้อมูลเข้าเป็นตัวเลขจำนวนเต็มหนึ่งจำนวน ผลการทำงานเป็นตัวเลขจำนวนเต็มหนึ่งจำนวนเช่นกัน รายละเอียดรหัสโปรแกรมของฟังก์ชั่นเป็นดังต่อไปนี้

```
Function Phi(N : Integer):Integer;
Var     I,Sum : Integer;
Begin
        Sum := 0;
        For I:=1 to N do
        If Gcd(I,N)=1 then Inc(Sum);
        Phi := Sum;
End;
```

4.7.3 ฟังก์ชั่นการตรวจสอบว่าจำนวนที่เป็นค่าพารามิเตอร์ของฟังก์ชั่นเป็นจำนวนเฉพาะหรือไม่ ข้อมูลเข้าเป็นตัวเลขจำนวนเต็มที่ต้องการตรวจสอบ ผลการทำงานจะเป็น True หรือ False ว่าข้อมูลเข้าเป็นจำนวนเฉพาะหรือไม่ รายละเอียดรหัสโปรแกรมของฟังก์ชั่นเป็นดังต่อไปนี้

```
function isprime(num:integer):boolean;
var   I : integer;
      chk : boolean;
begin
       chk:=true;
       For I:=2 to Round(Sqrt(num)) do
        If (num mod I)=0 then chk:=false;
       isprime:=chk;
end;
```

4.8 โครงสร้างข้อมูลที่กำหนดพิเศษ

โครงสร้างข้อมูลที่กำหนดขึ้นมาเป็นพิเศษนี้ได้กำหนดขึ้นมาเพื่อใช้กับชุดโปรแกรมทฤษฎีจำนวนโดยเฉพาะ โดยจะมีการเรียกใช้งานในบางส่วนโปรแกรมย่อยที่มีการทำงานเกี่ยวกับตัวเลขจำนวนมากๆ

โครงสร้างข้อมูลนี้จะเป็นลักษณะของแถวลำดับของจำนวนเต็มที่กำหนดขอบเขตของข้อมูลซึ่งถือเสมือนเลขโดดประจำหลักต่างๆ ของจำนวนนั้นๆ เป็นการนำความรู้ในเรื่องพื้นฐานของระบบจำนวนมาประยุกต์ใช้ในชุดโปรแกรม ส่วนของการประกาศโครงสร้างข้อมูล แสดงดังต่อไปนี้

```
DataType : array [0..15] of 0..9;
```

ในการใช้งานนั้นค่าดัชนีระบุแถวลำดับจะเป็นเลขยกกำลังของสิบ เช่นหากต้องการกำหนดค่า 2,558,003,625 ให้กับโครงสร้างข้อมูลนี้

0	0	2	5	5	8	0	0	3	6	2	5

จากตัวอย่างนี้จะได้จำนวนดังกล่าวซึ่งมีที่มาจาก $(5x10^0) + (2x10^1) + (6x10^2) + (3x10^3) + (0x10^4) + (0x10^5) + (8x10^6) + (5x10^7) + (5x10^8) + (2x10^9)$

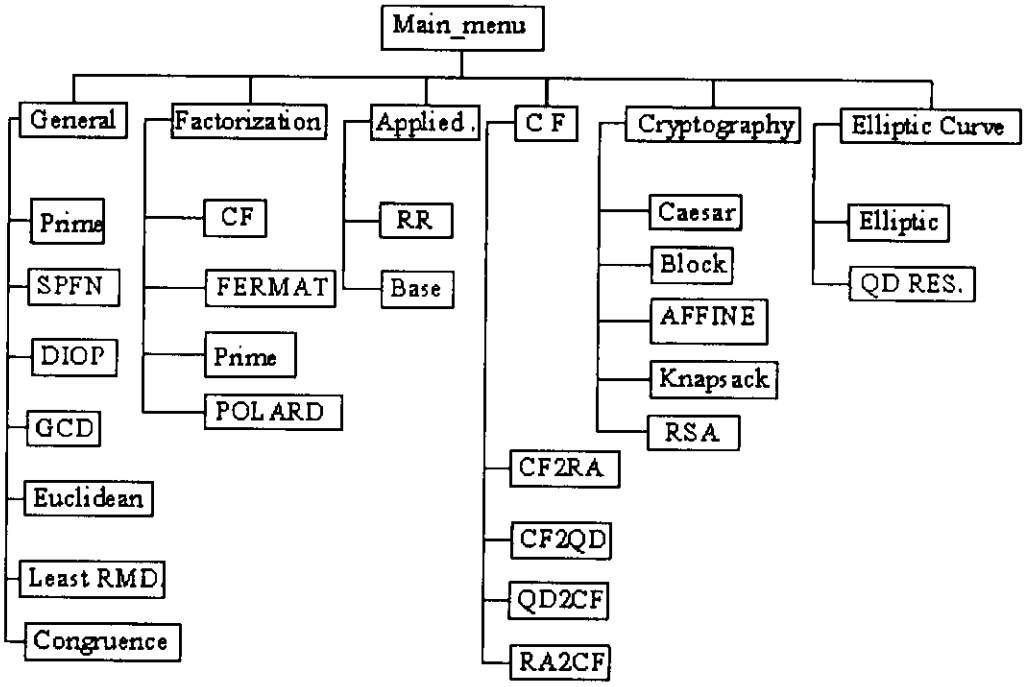
ซึ่งในส่วนนี้ยังจำเป็นต้องเรียกการใช้งานจากฟังก์ชันยกกำลังอีกด้วย รายละเอียดของฟังก์ชันดังนี้

```
function pow(x,y:integer):longint;
var j:integer;
   res:longint;
begin
   res:=1;
   for j:=1 to y do
      res:=res*x;
   pow:=res;
end;
```

จากนั้นจึงนำตัวเลขดังกล่าวนี้ไปคำนวณในหัวข้อต่างๆ ของวิชาทฤษฎีจำนวนต่อไป

4.9 การพัฒนาโปรแกรมสำหรับชุดโปรแกรมทฤษฎีจำนวน

จากการออกแบบชุดโปรแกรมทฤษฎีจำนวนที่กล่าวมาแล้วในหัวข้อ 4.2 – 4.8 ได้พัฒนาโปรแกรมที่มีโครงสร้างตามภาพประกอบที่ 4.12 และตารางที่ 4.1 แสดงหน้าที่โดยสรุปของแต่ละส่วนการทำงานหลักที่สำคัญ



ภาพประกอบ 4.12 โครงสร้างโปรแกรมสำหรับชุดโปรแกรมทฤษฎีจำนวน

ชื่อส่วนการทำงานหลัก / โมดูล	หน้าที่งานโดยสรุป
1. Main_menu	ส่วนโปรแกรมหลักที่เรียกการทำงานโมดูลย่อยต่างๆ
2. Prime	การหาจำนวนเฉพาะ
3. SPFN	ฟังก์ชันพิเศษต่างๆ
4. DIOP	สมการไดโอแฟนไทน์
5. GCD	การหาค่าตัวหารร่วมมาก
6. Euclidean	การหาค่าตัวหารร่วมมากโดยขั้นตอนวิธีของยูคลิเดียน
7. Least RMD.	การหาค่าตัวหารร่วมมากโดยขั้นตอนวิธีเศษเหลือน้อยที่สุด
8. Congruence	ระบบคอนกรูเอนซ์
9. CF	การแยกตัวประกอบโดยใช้เศษส่วนต่อเนื่อง
10. Fermat	การแยกตัวประกอบโดยวิธีของแฟร์มาต์
11. Prime	การแยกตัวประกอบ
12. Polard	การแยกตัวประกอบโดยวิธีของโพลาร์ด
13. RR	การสร้างตาราง Round- Robin
14. Base	การเปลี่ยนเลขฐาน
15. CF2RA	การแปลงเศษส่วนต่อเนื่องเป็นจำนวนตรรกยะ
16. CF2QD	การแปลงเศษส่วนต่อเนื่องเป็นจำนวนอตรรกยะ
17. QD2CF	การแปลงจำนวนอตรรกยะเป็นเศษส่วนต่อเนื่อง
18. RA2CF	การแปลงจำนวนตรรกยะเป็นเศษส่วนต่อเนื่อง
19. Caesar	การเข้ารหัสแบบซีซ่าร์
20. Block	การเข้ารหัสแบบบล็อค
21. Affine	การเข้ารหัสแบบแอฟไฟน์
22. Knapsack	การเข้ารหัสแบบแนบแสค
23. RSA	การเข้ารหัสแบบอาร์เอสเอ
24. Elliptic	Elliptic Curve
25. QD Res.	Quadratic Residue

ตาราง 4.1 ส่วนการทำงานหลักที่สำคัญในชุดโปรแกรมทฤษฎีจำนวน

4.10 **การทำงานติดต่อกับผู้ใช้** (User Interface)

การทำงานติดต่อกับผู้ใช้ของชุดโปรแกรมทฤษฎีจำนวนเป็นไปตามรายละเอียดและขั้นตอนที่แสดงไว้ในภาคผนวก ก. ซึ่งเป็นคู่มือการใช้ชุดโปรแกรมนี้