

บทที่ 3

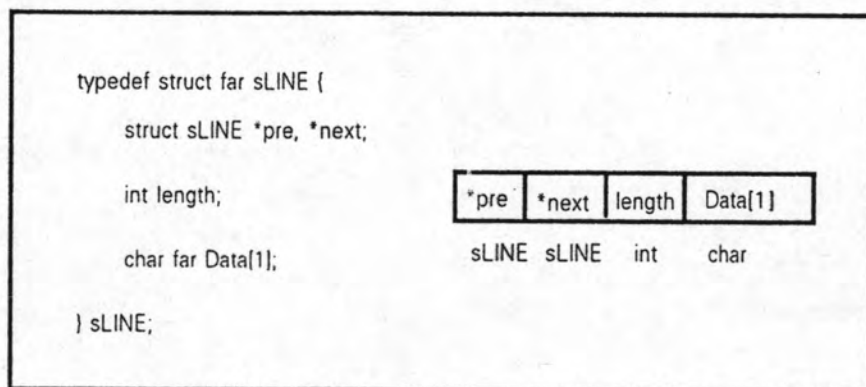
การออกแบบและพัฒนาโปรแกรม

ในการออกแบบและการพัฒนาโปรแกรม มีสิ่งที่ต้องพิจารณาหลายประการด้วยกัน ซึ่งสามารถแยกอธิบายเป็นส่วนๆ ได้ดังนี้

โครงสร้างข้อมูล (Data structure) (14)

โครงสร้างข้อมูลที่ใช้ในโปรแกรม ได้กำหนดให้ใช้โครงสร้างต่างๆกัน โดยในแต่ละโครงสร้าง มีหลักในการตั้งชื่อคือ ทุกโครงสร้างข้อมูลจะนำหน้าด้วยตัวอักษร s เป็นตัวพิมพ์เล็ก และตามด้วยชื่อของโครงสร้างข้อมูลเป็นตัวพิมพ์ใหญ่ ซึ่งโครงสร้างข้อมูลที่ใช้มีดังนี้

1. โครงสร้างของบรรทัด ชื่อว่า sLINE ดังรูปที่ 3.1 ในโครงสร้างนี้จะมีตัวชี้อยู่ 2 ตัวด้วยกัน โดยตัวแรกจะชี้ไปยังบรรทัดที่อยู่ก่อนหน้า และตัวชี้อีกตัวหนึ่งจะชี้ไปยัง บรรทัดที่อยู่ถัดไป โดยบรรทัดทั้งหมดจะเชื่อมกัน ด้วยโครงสร้างข้อมูลชนิดดับเบิลลิงค์เชอควูลิสต์

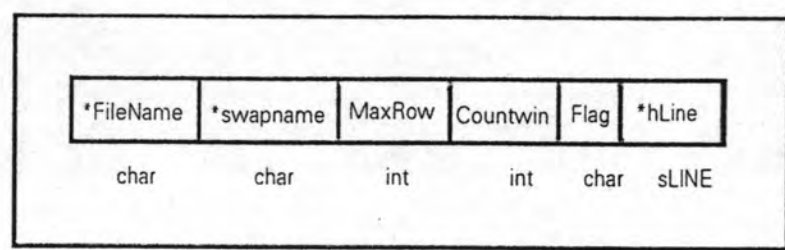


รูปที่ 3.1 แสดงโครงสร้างบรรทัด

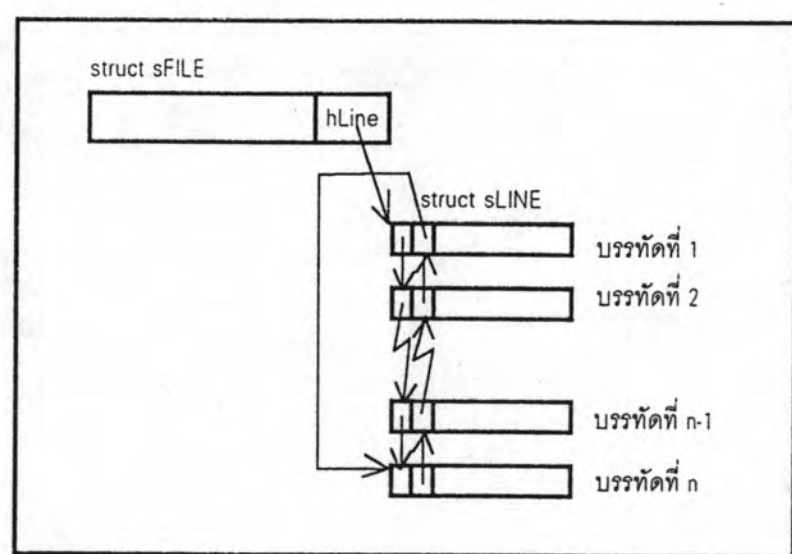
ในส่วนที่เก็บความยาว (length) จะเก็บความยาวจริงของข้อมูลในบรรทัดนั้น โดยนับตามจำนวนไบต์ที่ใช้จริง และไม่มีการชดเชยในกรณีที่มีการตั้งระยะ (Tab) ส่วนท้ายสุดจะเป็นส่วนของข้อมูล

จริงที่จะเก็บ Data[1] ในส่วนนี้จะกำหนดไว้เพียง 1 ไบต์เท่านั้น แต่ข้อมูลจริงๆอาจมีมากกว่าก็ได้ ตั้งแต่ 0 ถึง 512 ไบต์ ซึ่งโปรแกรมจะทำการขอจองหน่วยความจำ ตามจำนวนที่ต้องการใช้จริงอีกครั้งหนึ่ง ข้อมูลที่เก็บในส่วนนี้สามารถเก็บรหัสได้ทุกชนิด โดยมีค่าของรหัสตั้งแต่ 0 ถึง 255 โครงสร้างนี้ถือว่าเป็นโครงสร้างพื้นฐานของการทำงานในโปรแกรมทั้งหมด

2. โครงสร้างของแฟ้ม ชื่อว่า sFILE ดังรูปที่ 3.2 ในโครงสร้างแฟ้มนี้จะมีตัวชี้ตัวสุดท้ายที่ทำการชี้ไปที่บรรทัดแรกของแฟ้ม ทำให้ดูเหมือนว่าข้อมูลทั้งแฟ้ม คือทุกบรรทัดจะมาเชื่อมต่อกับแฟ้มที่จุดนี้ ดังรูปที่ 3.3 โครงสร้างนี้จะมีการกำหนดขึ้นจำนวนเท่ากับจำนวนแฟ้มที่ทำการเปิดอยู่ และให้นับแฟ้มที่เป็นแฟ้มเดียวกัน แต่ทำการเปิดหลายครั้ง ให้นับเป็น 1 แฟ้มเท่านั้น



รูปที่ 3.2 แสดงโครงสร้างของแฟ้ม



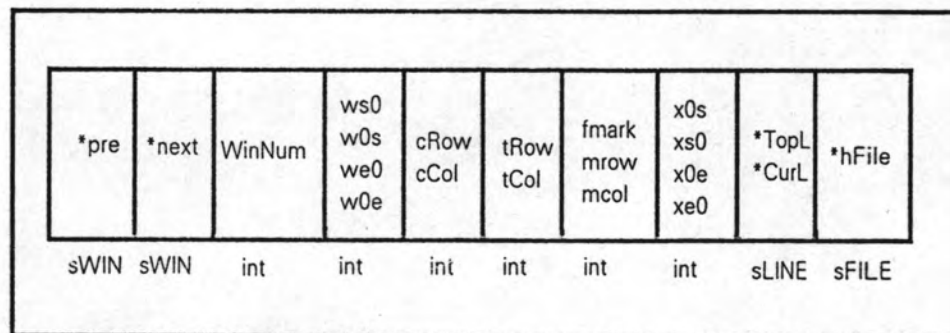
รูปที่ 3.3 แสดงโครงสร้างบรรทัดในการเชื่อมต่อกับโครงสร้างแฟ้ม

```

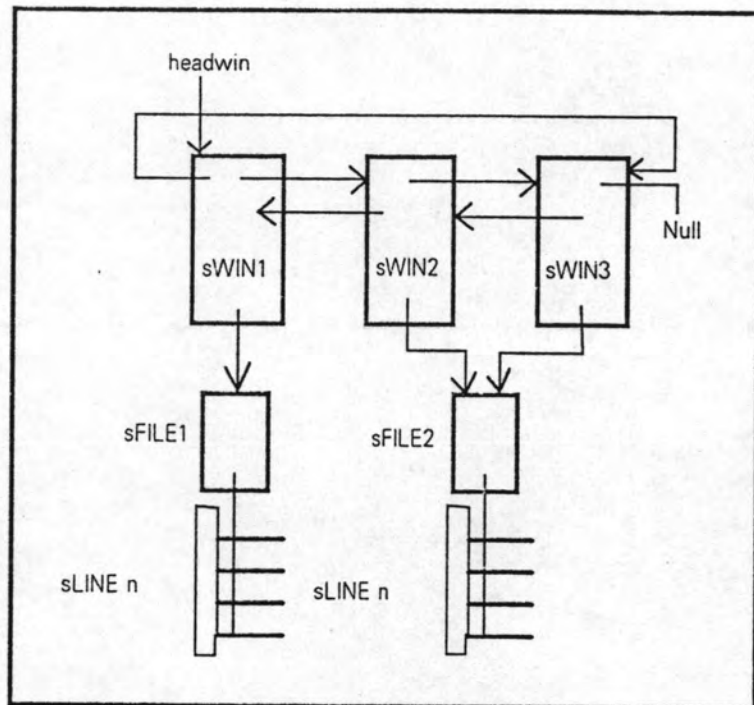
typedef struct far sFILE { // กำหนดชื่อของโครงสร้างแฟ้มเป็น "sFILE"
    char *FileName; // เก็บชื่อแฟ้มที่ทำการเปิดอยู่
    char *swapname; // เก็บชื่อแฟ้มที่ทำการสลับมอดูล
    int MaxRow; // เก็บว่าแฟ้มมีกี่บรรทัด
    char flag; // เก็บสถานะที่เกี่ยวกับแฟ้ม
    sFILE *hfile; // ตัวชี้ที่ทำการชี้ไปที่บรรทัดแรกสุดของแฟ้ม
} sFILE;

```

3. โครงสร้างของวินโดว์ ชื่อว่า sWIN ดังรูปที่ 3.4 ในส่วนนี้จะเก็บข้อมูลที่เกี่ยวข้องกับการเปิดวินโดว์และจะมีการกำหนดขึ้นเท่ากับจำนวนวินโดว์ที่ทำการเปิดอยู่ทั้งหมดในขณะที่ทำงานอยู่ จนกว่าจะมีการปิดวินโดว์ โครงสร้างของวินโดว์จะเชื่อมต่อกันแบบดับเบิลลิงค์เชอคู่อลิสต์ ดังรูปที่ 3.5 และในส่วนท้ายของโครงสร้างวินโดว์จะเป็นตัวชี้ที่ทำการชี้ไปยังโครงสร้างแฟ้ม ในระบบของโปรแกรมที่พัฒนาขึ้น จะมีตัวชี้ชื่อว่า headwin ที่จะทำการชี้ไปยังโครงสร้างของวินโดว์แรกสุด



รูปที่ 3.4 แสดงโครงสร้างข้อมูลวินโดว์



รูปที่ 3.5 แสดงโครงสร้างวินโดว์เชื่อมต่อกับโครงสร้างแฟ้ม

```

typedef struct far sWIN {
    struct sWIN *pre,*next; // กำหนดชื่อของโครงสร้างวินโดว์ "sWIN"
    int WinNum; // ตัวชี้ไปยังวินโดว์ก่อนและหลังตามลำดับ
    int ws0, w0s, we0, w0e; // เก็บลำดับที่ในการเปิดวินโดว์
    int cRow, cCol; // เก็บตำแหน่งกรอบของวินโดว์
    int tRow, tCol; // เก็บตำแหน่งของตัวชี้ตำแหน่ง
    int fmark; // เก็บตำแหน่งของข้อมูลเริ่มต้นวินโดว์
    int mrow, mcol, xs0, x0s, xe0, x0e; // เก็บสถานะบล็อก
    sLINE *topL, *Curl; // ตำแหน่งการกำหนดบล็อก
    sFILE *hFile; // ตัวชี้ไปบรรทัดแรกของวินโดว์และตัวชี้บรรทัดปัจจุบัน
} sWIN; // ตัวชี้ไปยังโครงสร้างแฟ้ม

```

4. โครงสร้างคำสั่ง ชื่อว่า sCOM ใช้ในการเก็บความสัมพันธ์ระหว่าง เลขที่ กระบวนการทำงาน และรหัสการกดแป้นพิมพ์

```
typedef struct far sCOM {           // กำหนดชื่อของโครงสร้างคำสั่ง
    unsigned char far mod;         // เก็บภาวะการกดแป้นพิมพ์
    unsigned char far k[2];       // เก็บรหัสสแกนโค้ด (Scan code)
} sCOM;
```

5. โครงสร้างแมโคร ชื่อว่า sMAC ใช้ในการเก็บคำสั่งแมโคร

```
typedef struct sMAC {              // กำหนดชื่อของโครงสร้าง
    struct sMAC *next;            // ตัวชี้ไปยังแมโครถัดไป
    int length;                   // จำนวนไบต์ที่ใช้
    uKEY mkey;                    // รหัสแป้นพิมพ์ที่ใช้เรียกแมโคร
    uKEY Data[1];                // ข้อมูลแมโคร
} sMAC;
```

6. โครงสร้าง sDELBUFF เป็นโครงสร้างที่ใช้ในการเก็บข้อมูลที่ถูกลบ เนื่องจากการใช้คำสั่งต่างๆในการลบข้อมูลออกจากพื้นที่ทำงานหลัก

```
typedef struct sDELBUFF {         // กำหนดชื่อของโครงสร้าง
    int length;                  // เก็บขนาดความยาวของข้อมูล
    int r,c;                     // บรรทัด และ คอลัมน์ ที่ลบ
    char *pc;                    // ตัวชี้ตำแหน่งแรกของข้อมูล
} sDELBUFF;
```

7. โครงสร้าง sSCRAB เป็นโครงสร้างที่ใช้ในการเก็บข้อมูล ของคลิปบอร์ด และสแครป (Scrap) มีโครงสร้างดังนี้

```
typedef struct sSCRAB {          // กำหนดชื่อของโครงสร้าง
    struct sSCRAB *next;        // ตัวชี้ไปยังสแครปต่อไป
    int fcl;                    // สถานะสแครป
    char flin;                  // สถานะของบรรทัดที่เก็บ
    char *name;                // ชื่อของสแครป
```

```

        sLINE *h;                // ตัวชี้ไปยังบรรทัดแรกสุด
    } sSCRAB;

```

8. โครงสร้างรายการเลือก ชื่อว่า sMENU เป็นโครงสร้างที่ใช้ในการสร้างรายการเลือก มีโครงสร้างดังนี้คือ

```

typedef struct sMENU {          // กำหนดชื่อของโครงสร้าง
    int n;                      // เลขที่แถวปัจจุบัน
    int l;                      // จำนวนแถวทั้งหมด
    int w;                      // ความกว้างมากที่สุดของรายการเลือก
    int r, c;                   // ตำแหน่งที่จะแสดงบนจอภาพ
    int *f;                    // ตัวชี้ไปยังฟังก์ชันในการทำงาน
    int *h;                    // ตัวชี้ไปยังที่เก็บตำแหน่งแสดง Hot key
    int *k;                    // แถวลำดับเก็บ Hot key
    int **i;                   // ตัวชี้ไปยังที่เก็บข้อความที่จะแสดงในรายการเลือก
    sSCR *s;                   // ตัวชี้ไปยังพื้นที่เก็บจอภาพที่จะแสดงรายการเลือก
    struct sMENU **p;         // ตัวชี้ไปยังรายการเลือกลำดับต่อไป
} sMENU;

```

9. โครงสร้างความช่วยเหลือ ชื่อว่า sHELP เป็นโครงสร้างที่ใช้สร้างรายการความช่วยเหลือ มีโครงสร้างดังนี้

```

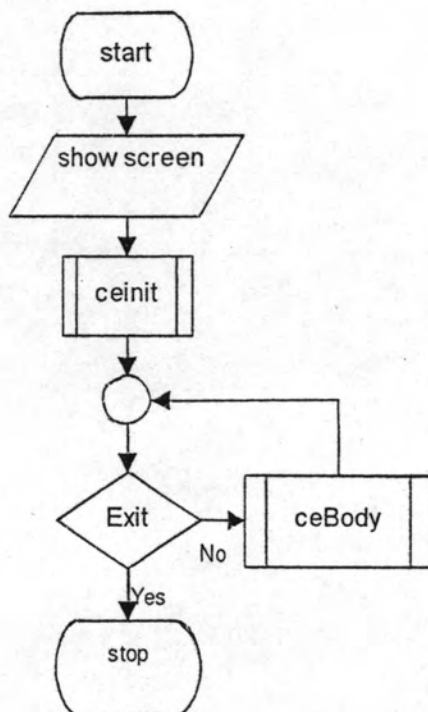
typedef struct sHELP {        // กำหนดชื่อโครงสร้าง
    int n;                    // จำนวนรายการ
    char **f;                 // ตัวชี้ไปยังที่เก็บชื่อฟังก์ชัน
    char **m;                 // ที่เก็บสถานะของแป้นพิมพ์
    sCOM *s0, *s1;           // ที่เก็บรหัสแป้นพิมพ์ 1 และ 2
} sHELP;

```

การทำงานของโปรแกรม

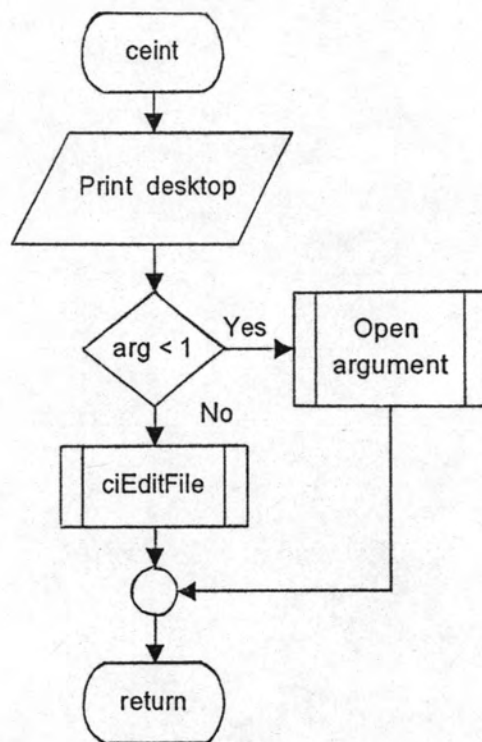
การทำงานของโปรแกรม สามารถแบ่งพิจารณาโครงสร้างหลักได้ดังนี้

1. มอดูลหลัก (Main) การทำงานของโปรแกรมจะเริ่มต้นจากมอดูลหลัก โดยมอดูลนี้จะมีการทำงานตาม รูปที่ 3.6 ในมอดูลนี้จะมีการเรียกฟังก์ชัน 2 ฟังก์ชันด้วยกัน คือ ceinit, ceBody



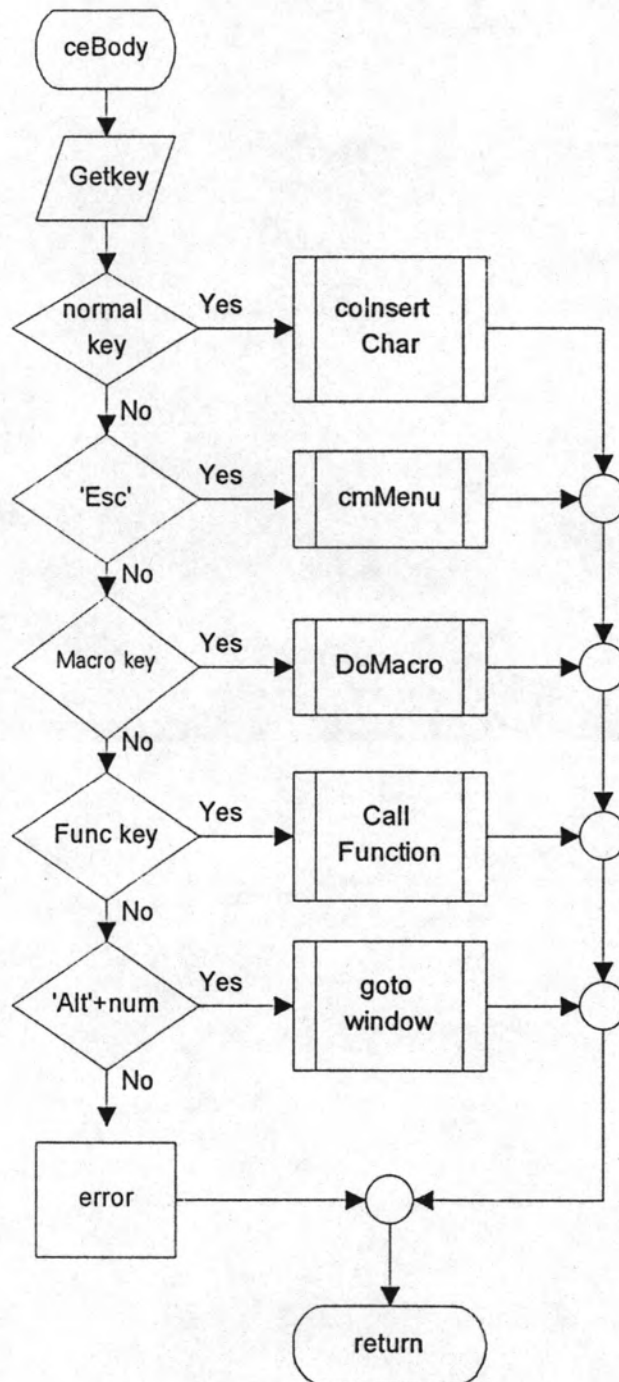
รูปที่ 3.6 แสดงการทำงานของมอดูลหลัก (Main)

2. ฟังก์ชัน ceinit มีการทำงาน ดังรูปที่ 3.7 ในฟังก์ชันนี้จะทำการติดตั้งค่าเริ่มต้นในการทำงานของโปรแกรมทั้งหมด และรับอาร์กิวเมนต์มาจากมอดูลหลัก เพื่อทำการเปิดแฟ้มหรือทำคำสั่งแอมโคร แต่ถ้าไม่มีอาร์กิวเมนต์ก็จะทำการเปิดแฟ้มข้อมูลใหม่



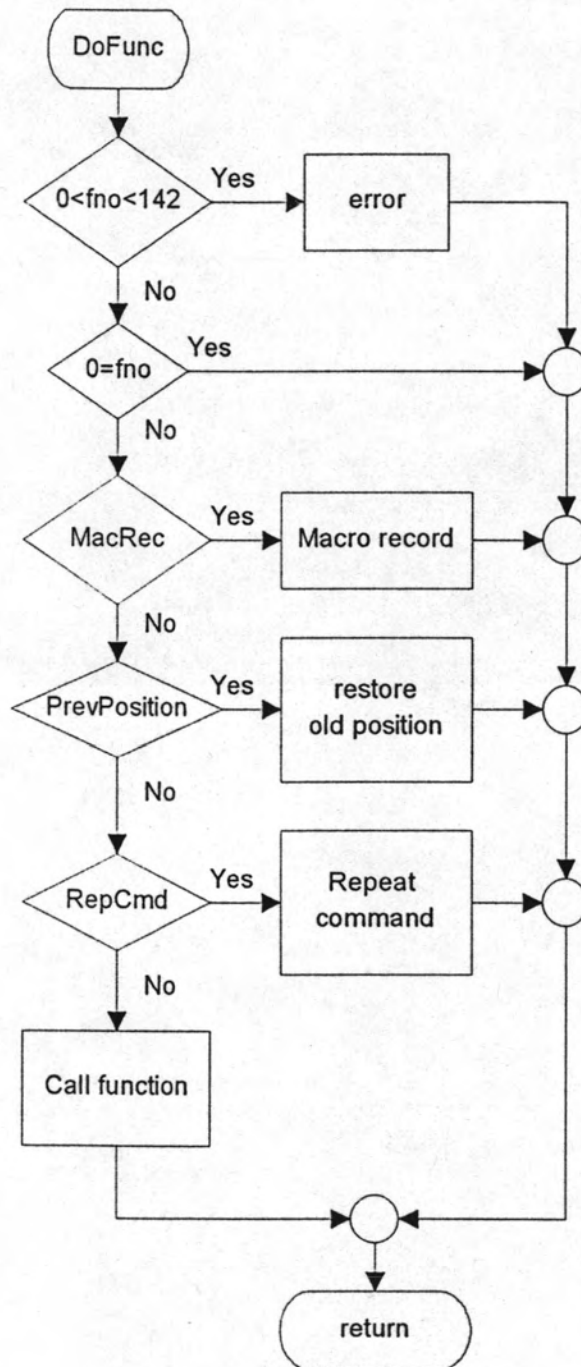
รูปที่ 3.7 แสดงการทำงานของฟังก์ชัน ceint

3. ฟังก์ชัน ceBody ฟังก์ชันนี้จะรวบรวมการทำงานทั้งหมด หลังจากมีการติดตั้งค่าเริ่มต้นต่างๆแล้ว แสดงดังรูปที่ 3.8 การทำงานจะเริ่มเมื่อมีการกดแป้นพิมพ์ แล้วฟังก์ชันนี้จะทำการคัดเลือกว่าจะทำงานใดต่อ การทำงานมีดังนี้คือ Insert char, menu, macro, function, goto windows, และ error



รูปที่ 3.8 แสดงการทำงานของฟังก์ชัน ceBody

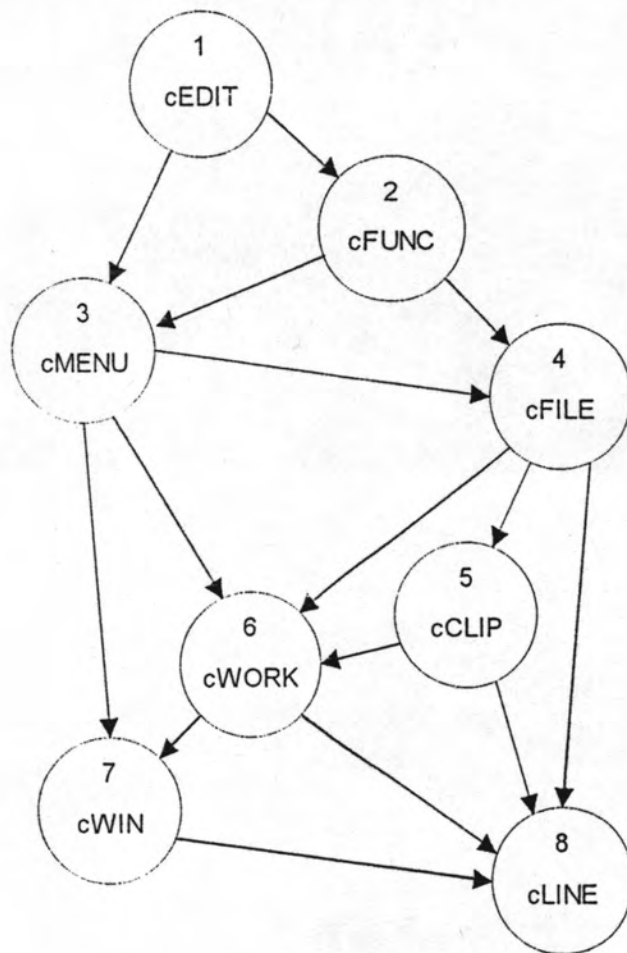
4 ฟังก์ชัน DoFunc หรือ "call function" เป็นฟังก์ชันซึ่งรับเลขที่ฟังก์ชันมา แล้วทำการแปลงให้เป็นตำแหน่งที่จะกระโดดไปทำงานตามเลขที่ฟังก์ชันนั้นๆ โดยมีการทำงานตามรูปที่ 3.9



รูปที่ 3.9 แสดงการทำงานฟังก์ชัน DoFunc

การแบ่งคลาส (Class division)

การทำงานของโปรแกรมทั้งหมด ถูกจัดให้เป็นกลุ่ม ในแต่ละกลุ่มถือว่าเป็นหนึ่งวัตถุ หรือเรียกว่าคลาส (Class) โดยมีคลาสอยู่ทั้งหมดจำนวน 8 คลาส ซึ่งแต่ละคลาสมีความสัมพันธ์กันดังรูปที่ 3.10 (รายละเอียดมีปรากฏในโปรแกรมภาคผนวก ก.)



รูปที่ 3.10 แสดงความสัมพันธ์ระหว่างคลาส

1. คลาส cEDIT เป็นคลาสเดียวที่มอดูลหลักรู้จัก เพราะเป็นคลาสที่รวมการทำงานทั้งหมดไว้ก่อนจะแยกไปทำงานย่อยในแต่ละคลาสอีกที ในคลาสนี้มีฟังก์ชันหลักอยู่ 2 ฟังก์ชันคือ ceinit และ ceBody สมาชิกของคลาสนี้จะเริ่มต้นด้วย ce (ภาคผนวก ก.1)

2. คลาส cFUNC เป็นคลาสที่เก็บรวบรวมฟังก์ชันการทำงานส่วนใหญ่ไว้ โดยมีฟังก์ชันหลักๆในการทำงานคือ กลุ่มฟังก์ชันแมโคร กลุ่มฟังก์ชันการเคลื่อนย้ายตัวชี้ตำแหน่ง (Cursor) กลุ่มฟังก์ชันการกระทำกับระเบียบ กลุ่มฟังก์ชันการขีดค่าและตั้งค่าต่างๆ สมาชิกของคลาสนี้จะขึ้นต้นด้วย cf (ภาคผนวก ก.2)

3. คลาส cMENU เป็นคลาสที่เกี่ยวกับรายการเลือก และรายการความช่วยเหลือ สมาชิกของคลาสนี้จะขึ้นต้นด้วย cm (ภาคผนวก ก.3)

4. คลาส cFILE เป็นคลาสที่มีการทำงานเกี่ยวกับการจัดการแฟ้มข้อมูล การจัดพิมพ์ และทำคำสั่งแมโคร สมาชิกของคลาสนี้จะขึ้นต้นด้วย ci (ภาคผนวก ก.4)

5. คลาส cCLIP เป็นคลาสที่จัดการเกี่ยวกับ คลิปบอร์ด (Clipboard) และสแครป (Scrap) สมาชิกของคลาสนี้จะขึ้นต้นด้วย cp (ภาคผนวก ก.5)

6. คลาส cWORK เป็นคลาสที่จัดการกับพื้นที่ทำงาน ซึ่งเป็นบรรทัดที่ถูกจองพื้นที่ไว้มากที่สุดเท่าที่บรรทัดหนึ่งๆจะมีได้ และการแก้ไขเปลี่ยนแปลงใดๆจะเกิดขึ้นที่บรรทัดนี้เท่านั้น บรรทัดนี้สามารถย้ายไปได้ ขึ้นอยู่กับว่าจะทำงานกับบรรทัดใด ก็จะทำให้การย้ายบรรทัดทำงานนี้ไปอยู่ที่บรรทัดนั้น บรรทัดนี้มีชื่อว่า wf สมาชิกของคลาสนี้จะขึ้นต้นด้วย co (ภาคผนวก ก.6)

7. คลาส cWIN เป็นคลาสที่จัดการกับการแสดงผลทางจอภาพ โดยการอ้างถึงตำแหน่งในหน่วยความจำแสดงผลโดยตรง ในคลาสนี้จะเกี่ยวกับการกำหนดบล็อก การแสดงผลของวินโดวที่เปิดอยู่ และไม่มีการเปลี่ยนแปลงข้อมูลของบรรทัดทำงานอยู่แต่อย่างใด สมาชิกของคลาสนี้จะขึ้นต้นด้วย cw (ภาคผนวก ก.7)

8. คลาส cLINE เป็นคลาสที่อยู่ต่ำสุด การทำงานจะมองเห็นแต่เฉพาะในระดับบรรทัดหนึ่งๆเท่านั้น และมีการจัดการกับการเก็บส่วนที่ได้ถูกลบไปแล้วด้วย คลาสนี้จะป็นคลาสสนับสนุนของคลาสอื่นอีกที สมาชิกของคลาสนี้จะขึ้นต้นด้วย cl (ภาคผนวก ก.8)

ฟังก์ชันในการทำงาน (Function)

ในการพัฒนาโปรแกรมบรรณาธิกรนี้ ได้แบ่งมอดูลการทำงานหลักออกเป็นฟังก์ชันย่อยๆ เพื่อให้สะดวกต่อการทำการพัฒนาโปรแกรมและแก้ไขข้อผิดพลาด ฟังก์ชันเหล่านี้สามารถจำแนกตามลักษณะการทำงานได้ 3 กลุ่มคือ

1. ฟังก์ชันแบบครอบคลุม (Global function) เป็นฟังก์ชันที่ฟังก์ชันอื่นๆสามารถเรียกใช้ได้โดยไม่ต้องมีการประกาศไว้ก่อน ฟังก์ชันในกลุ่มนี้ได้แก่

1.1 DoFunc เป็นฟังก์ชันสำหรับเรียกใช้ฟังก์ชันอื่น โดยจะรับเลขที่ของฟังก์ชันมาแล้วทำการแปลงเลขที่นั้นให้เป็นตำแหน่งของฟังก์ชัน แล้วส่งการควบคุมการทำงานไปทำงานต่อในตำแหน่งที่คำนวณได้ หรือมองอีกแง่หนึ่งคือการกระโดดไปทำงานตามเลขที่ของฟังก์ชันที่รับมา

1.2 err เป็นฟังก์ชันในการแสดงข้อความผิดพลาด

1.3 GetKey เป็นฟังก์ชันในการรับการกดแป้นอักขระ แล้วแปลงรหัส ที่ได้ให้อยู่ในรูปของค่าจำนวนเต็มเพื่อใช้งานต่อไป

1.4 MacReckey เป็นการบันทึกรหัสการกดแป้นอักขระไว้ในชุดคำสั่งแมโคร

1.5 ShowStatus เป็นฟังก์ชันที่จะแสดงผลในบรรทัดบนสุดของจอภาพ เกี่ยวกับสถานะในการทำงานขณะนั้น

1.6 ClearStatus เป็นฟังก์ชันลบบรรทัดบนสุดของจอภาพ ในบรรทัดแสดงสถานะการทำงาน

1.7 main เป็นฟังก์ชันหลักในการทำงานทั้งหมด

2. ฟังก์ชันชนิดเผยแพร่ได้ (Public function)

ฟังก์ชันชนิดเผยแพร่ได้นี้ เป็นฟังก์ชันที่เป็นสมาชิกของคลาส และเป็นฟังก์ชันที่จะถูกเรียกใช้ได้โดยฟังก์ชัน main หรือฟังก์ชัน DoFunc ในฟังก์ชันชนิดนี้จะได้อธิบายโดยละเอียด อีกครั้งหนึ่งในบทที่ 4 ฟังก์ชันชนิดนี้มีดังนี้คือ

2.1 cfBeginFile ย้ายตัวชี้ไปยังต้นแฟ้ม

2.2 cfBeginLine ย้ายตัวชี้ไปต้นบรรทัด

2.3 cfEndFile ย้ายตัวชี้ไปท้ายแฟ้ม

- 2.4 cfEndLine ย้ายตัวชี้ไปท้ายบรรทัด
- 2.5 coCursorLeft ย้ายตัวชี้ไปทางซ้ายหนึ่งอักขระ
- 2.6 cfCursorRight ย้ายตัวชี้ไปทางขวาหนึ่งอักขระ
- 2.7 cfCursorUp ย้ายตัวชี้ขึ้นหนึ่งบรรทัด
- 2.8 cfCursorDown ย้ายตัวชี้ลงหนึ่งบรรทัด
- 2.9 cfGotoColumn ย้ายตัวชี้ไปยังคอลัมน์
- 2.10 cfGotoLine ย้ายตัวชี้ไปยังบรรทัด
- 2.11 cfGotoBlockBeg ย้ายตัวชี้ไปต้นบล็อก
- 2.12 cfGotoBlockEnd ย้ายตัวชี้ไปท้ายบล็อก
- 2.13 cwWordLeft ย้ายตัวชี้ไปต้นคำทางซ้าย
- 2.14 cwWordRight ย้ายตัวชี้ไปต้นคำทางขวา
- 2.15 cwPrevPosition ย้ายตัวชี้ไปตำแหน่งเดิม
- 2.16 cfBegScreen ย้ายตัวชี้ไปตำแหน่งแรกของจอภาพ
- 2.17 cfEndScreen ย้ายตัวชี้ไปปลายบรรทัดสุดท้ายของจอภาพ
- 2.18 cfScreenLeft เลื่อนข้อความในวินโดวไปทางซ้ายหนึ่งอักขระ
- 2.19 cfScreenRight เลื่อนข้อความในวินโดวไปทางขวาหนึ่งอักขระ
- 2.20 cwScrollUp เลื่อนข้อความในวินโดวขึ้นหนึ่งบรรทัด
- 2.21 cwScrollDown เลื่อนข้อความในวินโดวลงหนึ่งบรรทัด
- 2.22 cfPageUp เลื่อนข้อความขึ้นหนึ่งวินโดว
- 2.23 cfPageDown เลื่อนข้อความลงหนึ่งวินโดว
- 2.24 cwMakeBotofScreen เลื่อนบรรทัดปัจจุบันมาอยู่บรรทัดล่างสุดของวินโดว
- 2.25 cwMakeCtrofScreen เลื่อนบรรทัดปัจจุบันมาอยู่กึ่งกลางวินโดว
- 2.26 cwMakeTopofScreen เลื่อนบรรทัดปัจจุบันมาอยู่บรรทัดบนสุดของวินโดว
- 2.27 cfNextPara ย้ายตัวชี้ไปวางต้นย่อหน้าถัดไป
- 2.28 cfPrevPara ย้ายตัวชี้ไปวางต้นย่อหน้าก่อนหน้าย่อหน้าปัจจุบัน
- 2.29 cfEndPara ย้ายตัวชี้มาวางท้ายย่อหน้า
- 2.30 ciNewFile สร้างแฟ้มใหม่ โดยไม่ถามชื่อแฟ้ม
- 2.31 ciEditFile สร้างแฟ้มใหม่ โดยถามชื่อแฟ้ม

- 2.32 ciKillFile ลบชื่อแฟ้มจากหน่วยบันทึก
- 2.33 ciSaveAs บันทึกแฟ้มเป็นชื่ออื่น
- 2.34 ciSaveFile บันทึกแฟ้มที่เปิดอยู่
- 2.35 ciFile ปิดแฟ้มที่เปิดอยู่ทั้งหมดและเตือนถ้ามีแฟ้มที่ถูกแก้ไขรวมอยู่ด้วย
- 2.36 ciPQuit ปิดแฟ้มที่เปิดอยู่ทั้งหมดโดยไม่มีการบันทึกข้อมูลใดๆ
- 2.37 ciExit ปิดแฟ้มที่เปิดอยู่ทั้งหมดและถามก่อนที่จะบันทึกแฟ้มที่ถูกแก้ไข
- 2.38 ciGSave บันทึกแฟ้มทั้งหมดที่เปิดอยู่ลงหน่วยบันทึก
- 2.39 ciGFile บันทึกแฟ้มที่ถูกแก้ไขทั้งหมดและออกจากโปรแกรม
- 2.40 ciGPQuit ออกจากโปรแกรมโดยไม่มีการบันทึก
- 2.41 ciGExit จะถามให้ทำการบันทึกแฟ้มที่ถูกแก้ไขและออกจากโปรแกรม
- 2.42 cfGetPrev สำเนาอักขระที่อยู่ในบรรทัดเหนือบรรทัดปัจจุบันมาแทรกที่ตำแหน่งตัวชี้
- 2.43 coBackspace ลบอักขระหนึ่งตัวทางซ้ายของตัวชี้
- 2.44 coDelCh ลบอักขระหนึ่งตัวที่ตำแหน่งตัวชี้
- 2.45 coDelLtWord ลบคำทางซ้ายหนึ่งคำ
- 2.46 coDelRtWord ลบคำทางขวาหนึ่งคำ
- 2.47 coUnKill ยกเลิกการลบหนึ่งครั้ง
- 2.48 coLiteral แทรกอักขระพิเศษในแฟ้ม
- 2.49 coUndoCursorLine ยกเลิกการเปลี่ยนแปลงในบรรทัดปัจจุบัน
- 2.50 coTabLt ย้ายตัวชี้ไปทางซ้ายหนึ่งระยะตั้ง
- 2.51 coTabRt ย้ายตัวชี้ไปทางขวาหนึ่งระยะตั้ง
- 2.52 cfAddLine เพิ่มบรรทัดว่างใต้บรรทัดปัจจุบัน
- 2.53 cfInsertLine เพิ่มบรรทัดว่างเหนือบรรทัดปัจจุบัน
- 2.54 cfDupLine สำเนาบรรทัดปัจจุบัน
- 2.55 coDelLine ลบบรรทัดปัจจุบัน
- 2.56 coDelToEol ลบข้อความตั้งแต่ตำแหน่งตัวชี้ไปสุดบรรทัด
- 2.57 coJoinLine รวมบรรทัดใต้บรรทัดปัจจุบันกับบรรทัดปัจจุบัน
- 2.58 coSplitLine ย้ายข้อความหลังตัวชี้ไปแทรกบรรทัดใหม่

- 2.59 coReturn ขึ้นบรรทัดใหม่
- 2.60 cfMarkBlockBegin กำหนดจุดเริ่มต้นบล็อก
- 2.61 cfMarkBlockEnd กำหนดจุดท้ายบล็อก
- 2.62 cwDropAnchor กำหนดบล็อกแบบเลื่อนได้โดยนับตำแหน่งตัวชี้เป็นส่วนหนึ่งของบล็อกด้วย
- 2.63 cwMarkCharacter กำหนดบล็อกแบบเลื่อนได้แต่ไม่นับตำแหน่งของตัวชี้เข้าเป็นส่วนหนึ่งของบล็อก
- 2.64 cwMarkColumn กำหนดบล็อกแบบคอลัมน์
- 2.65 cwMarkLine กำหนดบล็อกแบบบรรทัด
- 2.66 coMarkWord กำหนดบล็อกแบบคำ
- 2.67 cwUnmarkBlock ยกเลิกการบล็อก
- 2.68 cpCopyBlock ทำสำเนาบล็อกไปแทรกยังตำแหน่งตัวชี้
- 2.69 cpCopyOverBlock ทำสำเนาบล็อกไปทับตำแหน่งตัวชี้
- 2.70 cpMoveBlock ย้ายบล็อกไปตำแหน่งตัวชี้
- 2.71 cpDeleteBlock ลบบล็อก
- 2.72 ciReadBlock อ่านเพิ่มมายังตำแหน่งตัวชี้
- 2.73 ciWriteBlock บันทึกข้อความในบล็อกลงหน่วยบันทึก
- 2.74 cpFillBlock เติมอักขระในบริเวณที่มีการกำหนดบล็อก
- 2.75 cpShiftLeft เลื่อนข้อความในบล็อกไปทางซ้าย
- 2.76 cpShiftRight เลื่อนข้อความในบล็อกไปทางขวา
- 2.77 cpCut ย้ายข้อความในบล็อกไปเก็บไว้ในคลิปบอร์ด
- 2.78 cpCopy สำเนาข้อความในบล็อกไปเก็บไว้ในคลิปบอร์ด
- 2.79 cpPaste สำเนาข้อความในคลิปบอร์ดมาแทรกยังตำแหน่งตัวชี้
- 2.80 cpPasteOver สำเนาข้อความในคลิปบอร์ดมาทับตั้งแต่ตำแหน่งที่ตัวชี้อยู่
- 2.81 cpStoreScrbuff สำเนาข้อความในบล็อกไปเก็บไว้ในสแครบ
- 2.82 cpAppendScrbuff สำเนาข้อความในบล็อกไปต่อท้ายสแครบ
- 2.83 cpGetScrbuff สำเนาข้อความในสแครบมาตำแหน่งที่ตัวชี้อยู่
- 2.84 cpDelScrbuff ลบสแครบ

- 2.85 ciCurrentFilename กำหนดชื่อสแครป
- 2.86 ciExecuteScrap ทำงานตามคำสั่งในแมโคร
- 2.87 ciMacroRecord บันทึกคำสั่งแมโคร
- 2.88 ciMacroRead อ่านคำสั่งแมโครจากแฟ้ม
- 2.89 ciMacroWrite บันทึกแมโครลงในหน่วยบันทึก
- 2.90 ciMacroClear ลบแมโคร
- 2.91 ciPause หยุดการทำงานชั่วคราว
- 2.92 ciPrintAll พิมพ์แฟ้มออกทางเครื่องพิมพ์
- 2.93 ciPrintBlock พิมพ์ข้อความในบล็อกออกทางเครื่องพิมพ์
- 2.94 ciPrintEject เลื่อนกระดาษพิมพ์มาต้นหน้า
- 2.95 ciSetPrinterPort ติดตั้งช่องทางติดต่อเครื่องพิมพ์
- 2.96 ciSetPrintPageSize ตั้งขนาดกระดาษพิมพ์
- 2.97 ciSetPrintLeftMargin ตั้งระยะขอบซ้ายของการพิมพ์
- 2.98 ciSetPrintRightMargin ตั้งระยะขอบขวาของการพิมพ์
- 2.99 ciSetPrintTopMargin ตั้งระยะขอบบนของการพิมพ์
- 2.100 ciSetPrintBotMargin ตั้งระยะขอบล่างของการพิมพ์
- 2.101 ciPrevFile เปลี่ยนไปวินโดว์ก่อนหน้า
- 2.102 ciNextFile เปลี่ยนไปวินโดว์ถัดไป
- 2.103 ciWinNumber เปลี่ยนไปวินโดว์เลขที่
- 2.104 ciListAll แสดงวินโดว์ที่เปิดอยู่ทั้งหมด
- 2.105 ciRepaintDest พิมพ์จอภาพใหม่
- 2.106 ciSizeMove เปลี่ยนขนาดวินโดว์
- 2.107 cfZoomWindow เพิ่มขนาดวินโดว์ให้เต็มจอภาพ
- 2.108 cfShowEntryScreen แสดงหน้าจอก่อนการทำงานของโปรแกรม
- 2.109 ciCloseWindow ปิดวินโดว์
- 2.110 coSort จัดลำดับข้อความ
- 2.111 coFind ค้นหาข้อความ
- 2.112 coFindReplace ค้นหาข้อความและแทนที่ด้วยข้อความใหม่

- 2.113 coRepeatFind ค้นหาข้อความเดิม
- 2.114 cfLower เปลี่ยนอักษรให้เป็นตัวพิมพ์เล็ก
- 2.115 cfUpper เปลี่ยนอักษรให้เป็นตัวพิมพ์ใหญ่
- 2.116 cfFlip เปลี่ยนอักษรตัวพิมพ์ใหญ่ให้เป็นตัวพิมพ์เล็กและเปลี่ยนอักษรตัวพิมพ์เล็กให้เป็นตัวพิมพ์ใหญ่
- 2.117 cfCenterLine จัดข้อความให้อยู่กลางบรรทัด
- 2.118 cfLeftLine จัดข้อความชิดซ้าย
- 2.119 cfRightLine จัดข้อความชิดขวา
- 2.120 cfSetCTabWidth เปลี่ยนการตั้งระยะตัวชี้
- 2.121 cfSetPTabWidth เปลี่ยนการตั้งระยะกายภาพ
- 2.122 cfSetRmargin กำหนดขอบขวา
- 2.123 cfToggleSmartTabs เปลี่ยนภาวะการตั้งระยะ
- 2.124 cfToggleTabsExpand เปลี่ยนภาวะการขยายอักขระตั้งระยะ
- 2.125 cfToggleTabsOut เปลี่ยนภาวะการขยายการตั้งระยะ
- 2.126 cfToggleBackup เปลี่ยนภาวะการทำสำเนา
- 2.127 cfToggleBoxDraw เปลี่ยนภาวะการวาดกรอบ
- 2.128 cfToggleEnterMatching เปลี่ยนภาวะการจับคู่วงเล็บ
- 2.129 cfToggleIndent เปลี่ยนภาวะการย่อหน้า
- 2.130 cfToggleInsert เปลี่ยนภาวะการพิมพ์แทรกกับพิมพ์ทับ
- 2.131 cfToggleWordwrap เปลี่ยนภาวะการตัดคำ
- 2.132 cfToggleAddFormfeed เปลี่ยนภาวะการขึ้นหน้าใหม่ในการพิมพ์ทางเครื่องพิมพ์
- 2.133 cfToggleSwap เปลี่ยนภาวะการสลับข้อมูลลงหน่วยบันทึก
- 2.134 ciQuit ออกจากโปรแกรมโดยไม่มีการบินทิ้ง
- 2.135 cmMenu รายการเลือก
- 2.136 cmHelp รายการแสดงความช่วยเหลือ
- 2.137 cfRepeatCmd ทำคำสั่งล่าสุด
- 2.138 cfDos ทำคำสั่งดอส

- 2.139 cfShell ไประบบปฏิบัติการชั่วคราว
- 2.140 cfMatch หวางเล็บที่เข้าคู่กัน
- 2.141 coWrapPara ตัดช่องว่างที่อยู่ภายในย่อหน้า
- 2.142 colInsertChar ใส่อักขระที่ตำแหน่งตัวชี้
- 2.143 Jump คำสั่งแมโครให้กระโดดไปทำคำสั่งอื่น
- 2.144 JumpTrue คำสั่งแมโครให้กระโดดเมื่อแฟกซ์ของระบบเป็นจริง
- 2.145 JumpFalse คำสั่งแมโครให้กระโดดเมื่อแฟกซ์ของระบบเป็นเท็จ
- 2.146 ceinit การติดตั้งโปรแกรมเริ่มต้นครั้งแรก
- 2.147 ceBody การทำงานโปรแกรม หลังจากติดตั้งโปรแกรมแล้ว

3. ฟังก์ชันชนิดปกป้อง (Protected function)

ฟังก์ชันปกป้องเป็นฟังก์ชันที่เป็นสมาชิกของคลาสต่างๆ และถูกป้องกันหรือถูกอารักขา ภายใต้งี๋องไขของคลาซที่เป็นเจ้าของฟังก์ชันนั้น เมื่อมีการเรียกใช้จากฟังก์ชันในคลาซอื่น ฟังก์ชันในกลุ่มนี้มีดังน้คือ

- 3.1 clClearDelBuff ทำให้พื้นที่เก็บข้อความที่ลบว่าง
- 3.2 clAddDelBuff ใส่ข้อความที่ลบในพื้นที่เก็บข้อความการลบ
- 3.3 clNewLine การสร้างบรรทัดใหม่
- 3.4 clUpdateMark การปรับปรุงขอบเขตบล็อก
- 3.5 clCurlnMark ตรวจสอบว่าตัวชี้อยู่ในบล็อกหรือไม่
- 3.6 clDelLine ลบหนึ่งบรรทัด
- 3.7 clGotoLine ย้ายไปบรรทัดที่ต้องการ
- 3.8 clAddLine เพิ่มหนึ่งบรรทัด
- 3.9 clInsLine แทรกบรรทัด
- 3.10 clLogtoPhyCol คำนวณตำแหน่งข้อมูลจากตำแหน่งเสมือน
- 3.11 clPhytoLogCol คำนวณตำแหน่งข้อมูลจากตำแหน่งจริง
- 3.12 cwShowFilename แสดงชื่อแฟ้มที่กรอบของวินโดว์



- 3.13 cwColRowNum แสดงตำแหน่งแถวและคอลัมน์ที่กรอบของวินโดว์
- 3.14 cwScrollCol แสดงแถบการย้ายคอลัมน์
- 3.15 cwScrollRow แสดงแถบการย้ายแถว
- 3.16 cwCheckMark แสดงเครื่องหมายมีการแก้ไข
- 3.17 cwActFrame แสดงกรอบวินโดว์ให้เป็นกรอบใช้งาน
- 3.18 cwDeactFrame แสดงกรอบวินโดว์แบบซ่อน
- 3.19 cwShowLine แสดงหนึ่งบรรทัดทางจอภาพ
- 3.20 cwShowCursor แสดงตำแหน่งตัวชี้
- 3.21 cwShowDataScreen แสดงข้อความทั้งหน้าจอ
- 3.22 cwSaveFrame บันทึกตัวแปรในการทำงานเก็บไว้
- 3.23 cwRestoreFrame นำตัวแปรคืนการทำงานปกติ
- 3.24 cwSaveOnlyFrame เก็บอักขระเฉพาะที่เป็นกรอบ
- 3.25 cwRestoreOnlyFrame คืนอักขระเฉพาะที่เป็นกรอบ
- 3.26 cwClearStatus การลบบรรทัดสถานะโดยการกำหนดลักษณะเฉพาะ
- 3.27 cwShowAttLine แสดงสีของบรรทัด
- 3.28 cwShowAttScreen แสดงสีของทั้งวินโดว์
- 3.29 cwYesNoCancel ถามและให้เลือก Yes, No, Cancel
- 3.30 cwPositionCheckMark แสดงเครื่องหมายการแก้ไข
- 3.31 cwPrintDesk การพิมพ์หน้าจอใหม่
- 3.32 cwShowEof แสดงบรรทัดสิ้นสุดข้อมูล
- 3.33 cwShowSelect รายการถามและให้เลือก
- 3.34 cwShowSelectBox แสดงกรอบรายการถาม
- 3.35 cwCalCurFrameBox คำนวณตำแหน่งของกรอบ
- 3.36 cwCursor กำหนดการแสดงตัวชี้
- 3.37 cwStatusDispNum แสดงตัวเลข ที่บรรทัดแสดงสถานะ
- 3.38 cwDrawFrame แสดงกรอบ
- 3.39 cwPutString แสดงข้อความ
- 3.40 cwPickDefaultWord คำนวณตำแหน่งเริ่มต้นของคำที่สัมพันธ์กับตัวชี้

- 3.41 cwNumChar ตรวจสอบว่าเป็นตัวอักษรหรือตัวเลข
- 3.42 cwSaveWin บันทึกข้อมูลของวินโดว์เก็บไว้
- 3.43 cwRestoreWin นำข้อมูลของวินโดว์ออกมาใช้
- 3.44 cwStatusDisp แสดงข้อความที่บรรทัดแสดงสถานะ
- 3.45 coCurtoWork ทำบรรทัดที่ต้องการให้เป็นบรรทัดที่สามารถทำการแก้ไขได้
- 3.46 coShiftRightChar เลื่อนข้อความไปทางขวา
- 3.47 coShiftLeftChar เลื่อนข้อความไปทางซ้าย
- 3.48 coFreeWork ทำบรรทัดทำงานให้ว่าง
- 3.49 coJoinChar ต่อบรรทัดสองบรรทัดให้เป็นบรรทัดเดียวกัน
- 3.50 coInsertWord แทรกข้อความในบรรทัดทำงาน
- 3.51 coReplaceWord ทับข้อความในบรรทัดทำงาน
- 3.52 coUpdatePtr การสลับตัวชี้ของบรรทัดสองบรรทัด
- 3.53 cocheckSpace ตัดช่องว่างท้ายบรรทัด
- 3.54 coFindIndentCollum คำนวณตำแหน่งย่อหน้า
- 3.55 coMarkFind แสดงแถบสีในข้อความที่หาพบ
- 3.56 coReplaceFindWord แทนที่ข้อความที่ค้นหา
- 3.57 coDoOptions กำหนดทางเลือกในการค้นหาข้อความ
- 3.58 coGetNumber ถามและรับข้อมูลที่เป็นตัวเลข
- 3.59 coCheckAroundBox ตรวจสอบรอบตัวชี้ว่ามีอักขระที่เกี่ยวกับตารางหรือไม่
- 3.60 coUpperAndLower เปลี่ยนอักขระเป็นตัวพิมพ์ใหญ่หรือตัวพิมพ์เล็ก
- 3.61 coGetString รับข้อความจากผู้ใช้
- 3.62 coCompareLine เปรียบเทียบบรรทัด 2 บรรทัด
- 3.63 coBalanceLine1 คำนวณการจัดตำแหน่งข้อความ
- 3.64 coBalancedLine2 ย้ายข้อความมาตำแหน่งซ้ายหรือขวาหรือกลางของบรรทัด
- 3.65 coFindLine ค้นหาข้อความในบรรทัด
- 3.66 coCompareChar เปรียบเทียบอักขระ 2 ตัว
- 3.67 coWholeWordTest เปรียบเทียบคำ 2 คำ
- 3.68 cpFreeAll ทำข้อมูลในคลิปปอร์ดให้ว่าง และยกเลิกการบล็อก

- 3.69 cpAppen เพิ่มเติมข้อมูลในสแครป
- 3.70 cpDelMarkData ลบข้อความในบล็อกล
- 3.71 cpFindScrab ค้นหาสแครป
- 3.72 cpCopytoClip สำเนาบล็อกไปเก็บในคลิปบอร์ด
- 3.73 cpDelCurScrbuff ลบสแครป
- 3.74 cpSelectScrbuff เลือกลงสแครป
- 3.75 ciInsertTop ทำแฟ้มให้เป็นแฟ้มใช้งาน
- 3.76 ciSelectFile ค้นหาแฟ้ม
- 3.77 ciPrintMark พิมพ์ข้อความในบล็อกออกจากเครื่องพิมพ์
- 3.78 ciSelectFile แสดงรายการเลือกแฟ้มที่ต้องการเปิด
- 3.79 ciOpenNew เปิดแฟ้มใหม่
- 3.80 ciOpen เปิดแฟ้มเก่า
- 3.81 ciAddLine เพิ่มบรรทัดในแฟ้ม
- 3.82 ciNewsWin สร้างโครงสร้างวินโดว์
- 3.83 ciNewsFile สร้างแฟ้มใหม่
- 3.84 ciSaveFile บันทึกแฟ้มลงหน่วยบันทึก
- 3.85 ciMakeBackFile ทำการสำรองแฟ้มที่มีอยู่แล้ว
- 3.86 ciCloseTopWindow ปิดหนึ่งวินโดว์
- 3.87 ciCloseAll ปิดวินโดว์ทั้งหมด
- 3.89 ciPrevNextFile เลือกวินโดว์ก่อนหน้าหรือถัดไป
- 3.90 ciRepaintDesk พิมพ์หน้าจอใหม่
- 3.91 ciMakeFilename แปลงข้อความให้อยู่ในรูปของชื่อแฟ้ม
- 3.92 ciFindMac หาดำแหน่งเริ่มต้นคำสั่งแมโคร
- 3.93 ciFindTable หาเลขที่ฟังก์ชันที่จะทำงานต่อไป
- 3.94 ciMacroRead อ่านคำสั่งแมโครจากแฟ้ม
- 3.95 ciSwap การสลับข้อมูลในหน่วยความจำกับหน่วยบันทึก
- 3.96 ciDisktoMem นำข้อมูลในหน่วยบันทึกกลับสูหน่วยความจำหลัก
- 3.97 ciMemtoDisk บันทึกข้อมูลในหน่วยความจำหลักลงหน่วยบันทึก

- 3.98 ciMakeDir ทำข้อความให้เป็นไดเรกทอรีของชื่อแฟ้ม
- 3.99 ciGetFilename รับชื่อแฟ้ม
- 3.100 ciOpenFile เปิดแฟ้ม
- 3.101 cmShow แสดงตัวเลือกของรายการเลือก
- 3.102 cmClose ยุติรายการเลือก
- 3.103 cmShowH แสดงรายการย่อยข้อความช่วยเหลือ
- 3.104 cmMakeKey แปลงรหัสแป้นอักขระเป็นตัวอักษร
- 3.105 cfShell_Dos การออกจากโปรแกรมชั่วคราว

โปรแกรมต้นฉบับ (Source program)

โปรแกรมต้นฉบับของโปรแกรมบรรณาธิกร ประกอบด้วยโปรแกรมต่างๆ ดังตารางที่ 3.1

ชื่อโปรแกรม	จำนวนตัวอักษร	จำนวนบรรทัด
Class.hpp	15,508	394
Struct.hpp	2,201	103
Test.hpp	2,768	114
Menu.hpp	4,752	118
Extern.hpp	1,718	96
Help.hpp	4,698	106
main.cpp	35,334	815
cline.cpp	9,060	362
cwin.cpp	26,435	1,102
cwork.cpp	59,774	2,187
cfile.cpp	44,465	1,730
cclip.cpp	16,631	665
cmenu.cpp	15,063	508
test.cpp	149	8
cedit.cpp	3,067	110
cfunc.cpp	19,314	846
รวม	260,937	9,237

ตารางที่ 3.1 แสดงรายการโปรแกรมน้อย

การแปลโปรแกรม (Compile program)

โปรแกรมทั้งหมดถูกแยกเป็นฟังก์ชันต่างๆ โดยเริ่มต้นที่ฟังก์ชัน main ฟังก์ชันเหล่านี้จะถูกแบ่งเป็นกลุ่ม จำนวน 8 กลุ่มด้วยกันตามประเภทของฟังก์ชัน และยังมีฟังก์ชันที่เป็นแบบ

ครอบคลุมนอีก แต่ละกลุ่มจะเรียกว่าคลาสซึ่งแต่ละคลาสจะถูกเก็บแยกกันเป็นแฟ้ม มีชื่อดังนี้คือ cline.cpp, cwin.cpp, cwork.cpp, cclip.cpp, cfile.cpp, cmenu.cpp, cfunc.cpp, cedit.cpp ส่วนฟังก์ชันชนิดครอบคลุมนจะเก็บในอีกแฟ้มหนึ่งชื่อว่า main.cpp การแปลจะทำการแปลทีละแฟ้ม แล้วนำมาเชื่อมเข้าด้วยกัน ซึ่งจะได้โปรแกรมสุดท้ายเป็นโปรแกรมที่สามารถทำงานได้ และได้ตั้งชื่อโปรแกรมนี้ว่า โปรแกรมวังใหม่ (Wangmai) โดยมีชื่อสำหรับเรียกใช้งานว่า w.exe มีขนาด 158 กิโลไบต์