

## บทที่ 5

### การพัฒนาส่วนจำเพาะการค้นข้อความไทยในเอกสารพีดีเอฟ

การพัฒนาส่วนจำเพาะในเอกสารพีดีเอฟ จะใช้เครื่องมือในการพัฒนาร่วมกัน 2 ตัว คือ ไมโครซอฟต์วิซวลซีพลัสพลัส(Microsoft VisualC++) และชุดพัฒนาส่วนจำเพาะอะโครแบต การพัฒนาส่วนจำเพาะการค้นข้อความไทย จะเริ่มต้นด้วยการกำหนดคลาส ฟังก์ชัน และความสัมพันธ์ในการส่งข้อความติดต่อกันระหว่างฟังก์ชัน ขั้นตอนถัดไปจะพัฒนากระบวนการทำงานของแต่ละฟังก์ชัน

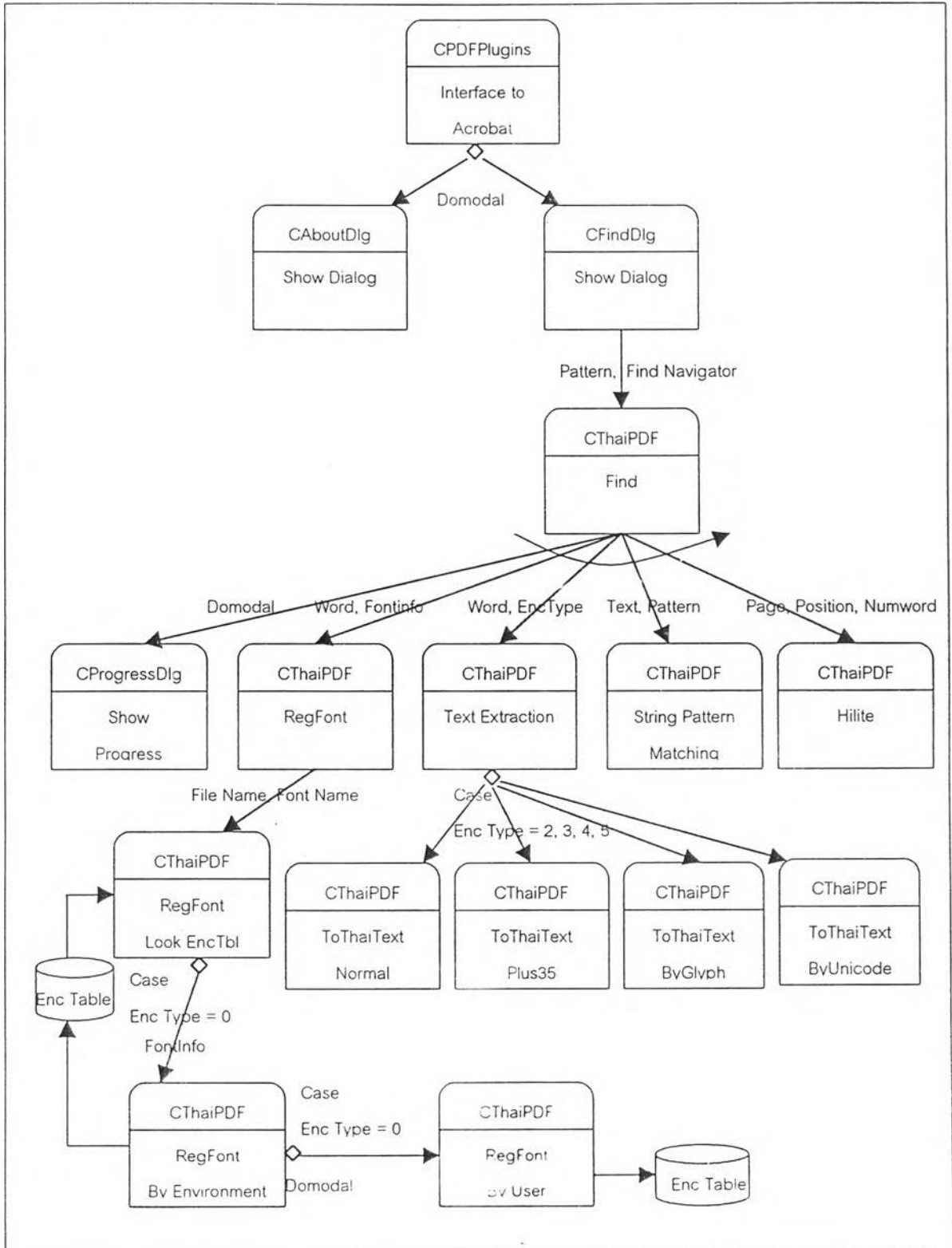
#### 5.1 ความสัมพันธ์ของคลาส(Class)ในส่วนจำเพาะการค้นข้อความไทยในเอกสารพีดีเอฟ

ในการพัฒนาส่วนจำเพาะการค้นข้อความไทยในเอกสารพีดีเอฟ ได้สร้างคลาสเพื่อใช้งานตามวัตถุประสงค์การทำงานขึ้นมา 6 คลาส ได้แก่

1. คลาส CPDFPlugins
2. คลาส CAboutDlg
3. คลาส CFindDlg
4. คลาส CThaiPDF
5. คลาส CProgressDlg
6. คลาส CFontRegDlg

ฟังก์ชันของแต่ละคลาสจะมีความสัมพันธ์และติดต่อกันตามหน้าที่ โดยมีกระบวนการทำงานดังนี้ (ดูรูปที่ 22 ประกอบ)

1. CPDFPlugins จะเชื่อมส่วนจำเพาะเข้ากับโปรแกรมแสดงเอกสารพีดีเอฟและติดต่อเรียกการทำงานกับส่วนจำเพาะ
2. ถ้ามีการส่งข้อความแสดงหน้าต่างวัตถุในคลาสCAboutDlg ให้สร้างวัตถุขึ้นมาแสดง
3. ถ้ามีการส่งข้อความแสดงหน้าต่างวัตถุในคลาสCFindDlg ให้สร้างวัตถุขึ้นมาแสดงเพื่อรอรับข้อความที่ต้องการค้น เมื่อผู้ใช้กดปุ่มค้น ข้อความที่ต้องการค้นและทิศทางจะถูกส่งไปให้ CThaiPDF::Find
4. CThaiPDF จะส่งข้อความให้ CProgressDlg สร้างวัตถุแสดงขณะที่กำลังทำการค้น
5. CThaiPDF::RegFont ทำการวิเคราะห์การเข้ารหัสแบบอักษร หากข้อมูลไม่เพียงพอในการวิเคราะห์ จะส่งข้อความให้CFontRegDlg::ByManual เพื่อสร้างวัตถุมาช่วยในการวิเคราะห์การเข้ารหัส
6. CThaiPDF::TextExtract จะทำการถอดรหัสข้อความไทยตามข้อมูลที่ได้จาก RegFont
7. CThaiPDF::ThaiPatternMatching รับข้อความที่ได้จากการถอดรหัสและค้นข้อความ
8. CThaiPDF::Hilite รับตำแหน่งและจำนวนข้อความมาแสดงสีที่ทับข้อความที่พบ



รูปที่ 22 ผังแสดงคลาสและความสัมพันธ์ในโปรแกรมสำหรับจัดการค้นหาข้อความไทยในเอกสารพีดีเอฟ

### การพัฒนาคลาสและฟังก์ชัน

คลาสที่ใช้ในการพัฒนาส่วนจำเพาะการค้นห้อยในเอกสารพีดีเอฟ มีฟังก์ชันการทำงานดังนี้

1. คลาส CPDFPlugins จะทำหน้าที่ในการให้บริการติดต่อส่วนจำเพาะกับโปรแกรมแสดงเอกสารพีดีเอฟ
2. คลาส CAboutDlg ทำหน้าที่แสดงข้อมูลเกี่ยวกับส่วนจำเพาะ เช่น รุ่น คำอธิบาย
3. คลาส CFindDlg ทำหน้าที่รับข้อความและทิศทางในการค้นห้อยข้อความ
4. คลาส CThaiPDF ทำหน้าที่จัดการเกี่ยวกับห้อยข้อความไทยในเอกสารพีดีเอฟ
5. คลาส CProgressDlg ทำหน้าที่ในการวิเคราะห์การเข้ารหัสแบบอักษร
6. คลาส CFontRegDlg ทำหน้าที่แสดงความก้าวหน้าขณะกำลังค้นห้อยข้อความ

(ดูรายละเอียดชุดคำสั่งของโปรแกรมเพิ่มเติมได้ที่ภาคผนวก ข)

#### คลาส CPDFPlugins

คลาสนี้ในส่วนจำเพาะการค้นห้อยข้อความไทยในเอกสารพีดีเอฟจะทำหน้าที่ในการให้บริการเชื่อมต่อส่วนจำเพาะเข้ากับโปรแกรมแสดงเอกสารพีดีเอฟ และเรียกใช้งานส่วนจำเพาะ ประกอบด้วยฟังก์ชันต่างๆ ดังนี้

PIHandshake() ทำหน้าที่ในการตรวจสอบการเชื่อมต่อส่วนจำเพาะกับโปรแกรมแสดงเอกสารพีดีเอฟ

```
CPDFPlugins::PIHandshake(ASUns32 handshakeVersion, void *handshakeData)
{
    IF รุ่นที่ใช้ในการตรวจสอบการเชื่อมต่อ เท่ากับ รุ่นที่0200 THEN
        ตรวจสอบประเภทของข้อมูลที่จะทำการเชื่อมต่อ
        กำหนดชื่อที่จะเรียกส่วนจำเพาะ
        ลงทะเบียนส่วนจำเพาะ
        Return true
    END IF
    Return false
}
```

InitMenuItem() ทำการเพิ่มเมนูคันไทยและปุ่มเครื่องมือคันไทย เข้าไปในกลุ่มเมนูและในกลุ่มเครื่องมือของโปรแกรมแสดงเอกสารพีดีเอฟ

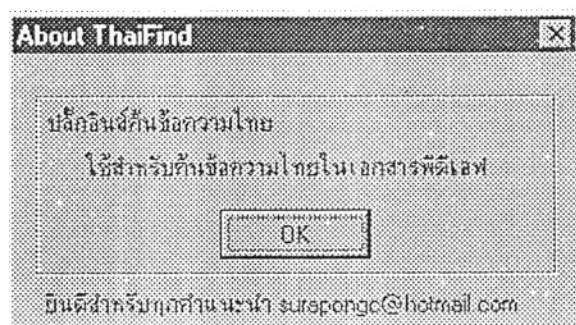
```
CPDFPlugins::InitMenuItem()
{
    ประกาศรูปปุ่มเครื่องมือและฟังก์ชันคันไทย
    เพิ่มรูปปุ่มเครื่องมือคันไทยเข้าไปในกลุ่มเครื่องมือ
    ปลดปล่อยปุ่มเครื่องมือที่ประกาศไว้ทั้งหมดออกจากหน่วยความจำ
    ประกาศเมนูและคำอธิบายของฟังก์ชันคันไทย
    เพิ่มเมนูคันไทยเข้าไปในเมนูของโปรแกรมแสดงเอกสารพีดีเอฟ
    ปลดปล่อยเมนูที่ประกาศไว้ทั้งหมดออกจากหน่วยความจำ
}
```

EnableMenuItem() อนุญาตให้เมนูทำงานได้เมื่อมีเอกสารเปิดอยู่ ถ้าไม่มีเอกสารเปิดอยู่ให้เมนูเป็นสีเทาจางและไม่อนุญาตให้เมนูทำงาน

```
CPDFPlugins::EnableMenuItem(void *data)
{
    IF มีเอกสารถูกเปิดอ่าน THEN
    {
        อนุญาตให้ใช้เมนูคันไทย
        IF มีการแสดงสีที่ทับข้อความ THEN
            ล้างสีที่บอกรอกจากข้อความ
        END IF
    }
    ELSE
    {
        กำหนดให้ตัวแปรที่เก็บข้อความที่จะคันเป็นค่าว่าง
        ไม่อนุญาตให้ใช้เมนูคันไทย โดยแสดงเป็นสีเทาจางและไม่รับคำสั่ง
    }
}
```

### คลาส CAboutDlg

ใช้ในการแสดงข้อมูลเกี่ยวกับส่วนจำเพาะการค้นข้อความไทยในเอกสารพีดีเอฟ เช่น รุ่น คำอธิบาย ผู้สร้าง และอื่นๆ (ดูรูปที่ 23 ประกอบ) ประกอบด้วยฟังก์ชันต่างๆดังนี้



รูปที่ 23 หน้าต่างของคลาส CAboutDlg

OnInitDialog() กำหนดค่าเริ่มต้นให้กับวัตถุของคลาส

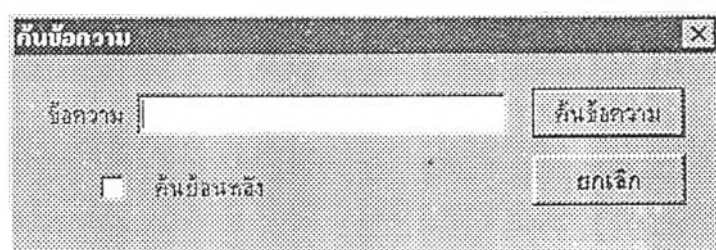
```

BOOL CAboutDlg::OnInitDialog()
{
    กำหนดค่าเริ่มต้นให้กับคลาส
    Return แสดงว่าเรียกหน้าต่างของคลาสนี้ได้สำเร็จ
}

```

### คลาส CThaiFindDlg

เมื่อผู้ใช้เรียกให้ส่วนจำเพาะค้นข้อความทำงาน คลาสนี้จะถูกเรียกโดยคลาส CPDFPlugins ขึ้นมาแสดงเพื่อรับข้อความที่ต้องการค้นจากผู้ใช้ (ดูรูปที่ 24 ประกอบ) ประกอบด้วยฟังก์ชันต่างๆดังนี้



รูปที่ 24 หน้าต่างของคลาส CFindDlg

CFindDlg() เป็นการกำหนดค่าเริ่มต้นโดยตัวคลาสเอง ซึ่งจะกำหนดค่าเริ่มต้นให้ช่องข้อความที่ต้องการค้นและทิศทางการค้นย้อนหลังเป็นช่องว่าง

```
CFindDlg::CFindDlg(CWnd* pParent /*=NULL*/)
: CDialog(CFindDlg::IDD, pParent)
{
    กำหนดให้ข้อความที่ต้องการค้นเป็นค่าว่าง
    ตัวเลือกการค้นย้อนหลังเป็นค่าว่าง
}
```

DoDataExchange() จะนำค่าที่ได้รับในองค์ประกอบของหน้าต่างในคลาสนี้ มาเก็บไว้ในตัวแปรที่เรากำหนด เช่น ข้อความที่ต้องการค้นจะถูกเก็บไว้ในตัวแปร m\_strPattern เป็นต้น

```
void CFindDlg::DoDataExchange(CDataExchange* pDX)
{
    นำค่าที่ได้จากช่องรับข้อความที่ต้องการค้นมาเก็บไว้ในตัวแปร m_strPattern
    นำค่าที่ได้จากตัวเลือกทิศทางการค้นมาเก็บไว้ในตัวแปร m_isBackward
}
```

OnCheckBackward() จะทำงานเมื่อมีการกดเลือกที่ตัวเลือกทิศทางการค้น โดยจะสลับค่าจริงหรือค่าเท็จให้กับตัวแปรที่แสดงทิศทางการค้น

```
Void CfindDlg::OnCheckBackward()
{
    IF ตัวแปร m_isBackward เป็นจริง THEN
        กำหนดให้ตัวแปร m_isBackward เป็นเท็จ
    ELSE
        กำหนดให้ตัวแปร m_isBackward เป็นจริง
    END IF
}
```

### คลาส CThaiPDF

คลาสนี้ในส่วนจำเพาะจะใช้ในการจัดการกับข้อความไทยในเอกสารพีดีเอฟ ประกอบด้วยฟังก์ชันต่างๆ ดังนี้

FindForward() คือ กระบวนการค้นข้อความไปข้างหน้า มีขั้นตอนการทำงานดังนี้

```
CThaiPDF::FindForward(Cstring pattern) {
    IF มีการแสดงสีทับข้อความในเอกสาร THEN
        ตำแหน่งเริ่มต้นในการค้นคือตำแหน่งหลังข้อความที่โดนสีทับ
    ELSE
        ตำแหน่งเริ่มต้นในการค้นคือข้อความแรกในเอกสารหน้านั้น
    END IF
    WHILE (ไม่สิ้นสุดเอกสาร และไม่พบข้อความ และ ผู้ใช้ไม่ได้ยกเลิกการค้น) {
        แสดงว่ากำลังค้นข้อความด้วยวัตถุคลาส CProgressDlg
        WHILE (ไม่หมดข้อความ และไม่พบข้อความ และ ผู้ใช้ไม่ได้ยกเลิกการค้น) {
            วิเคราะห์การเข้ารหัสด้วย CThaiPDF::RegFont()
            ถอดรหัสข้อความด้วย CThaiPDF::TextExtract()
            ค้นข้อความด้วย CThaiPDF::ThaiPatternMatching()
            เลื่อนข้อความถัดไป
        } END LOOP
        IF พบข้อความที่ต้องการค้น THEN
            แสดงสีทับข้อความที่ต้องการค้นด้วย CThaiPDF::Hilite()
        ELSE
            { เลื่อนหน้าเอกสารเป็นเอกสารหน้าถัดไป
              IF สิ้นสุดเอกสาร THEN
                  แสดงให้ผู้ใช้ทราบว่าสิ้นสุดเอกสารแล้ว
                  IF ผู้ใช้ยังต้องการค้นข้อความ THEN เลื่อนหน้าเอกสารไปที่หน้าแรก
                  ELSE ปิดการทำงาน
                END IF }
        END IF
    } END LOOP
}
```

```

FindBackward() คือ กระบวนการค้นข้อความย้อนหลัง มีขั้นตอนการทำงานดังนี้
CThaiPDF::FindBackward(Cstring pattern) {
    IF มีการแสดงสีที่ทับข้อความในเอกสาร THEN
        ตำแหน่งเริ่มต้นในการค้นคือตำแหน่งก่อนข้อความที่โดนสีที่ทับ
    ELSE
        ตำแหน่งเริ่มต้นในการค้นคือข้อความสุดท้ายในเอกสารหน้านั้น
    END IF
    WHILE (ไม่สิ้นสุดเอกสาร และ ไม่พบข้อความ และ ผู้ใช้ไม่ได้ยกเลิกการค้น) {
        แสดงว่ากำลังค้นข้อความด้วยวัตถุคลาส CProgressDlg
        WHILE (ไม่หมดข้อความ และ ไม่พบข้อความ และ ผู้ใช้ไม่ได้ยกเลิกการค้น) {
            วิเคราะห์การเข้ารหัสด้วย CThaiPDF::RegFont()
            ถอดรหัสข้อความด้วย CThaiPDF::TextExtract()
            ค้นข้อความด้วย CThaiPDF::ThaiPatternMatching()
            เลื่อนข้อความเป็นข้อความก่อนข้อความนี้
        } END LOOP
        IF พบข้อความที่ต้องการค้น THEN
            แสดงสีที่ทับข้อความที่ต้องการค้นด้วย CThaiPDF::Hilite()
        ELSE {
            เลื่อนหน้าเอกสารเป็นหน้าเอกสารก่อนหน้านี
            IF หน้าเอกสารเท่ากับศูนย์ THEN
                แสดงให้ผู้ใช้ทราบว่าสิ้นสุดเอกสารแล้ว
                IF ผู้ใช้ยังต้องการค้นข้อความ THEN
                    เลื่อนหน้าเอกสารไปที่หน้าสุดท้ายของเอกสาร
                ELSE
                    ปิดการทำงาน
                END IF
            }
        }
    } END IF
} END LOOP
}

```



RegFont() ทำการวิเคราะห์การเข้ารหัสตัวอักษรในเอกสารพีดีเอฟ ว่ามีการเข้ารหัสเป็นแบบใด มีการทำงานดังนี้

```
CThaiPDF::RegFont(PDWordFinder WordFinder, PDWord word, PDFont font)
{
    หาประเภทการเข้ารหัสแบบอักษรจากฐานข้อมูล
    ที่เก็บการเข้ารหัสที่เคยได้วิเคราะห์มาแล้ว
    IF ไม่พบการเข้ารหัสของแบบอักษร THEN {
        วิเคราะห์การเข้ารหัสตัวอักษรโดยใช้ข้อมูลเวดลุ่มต่างๆในเอกสาร
        IF ข้อมูลเวดลุ่มในเอกสารไม่เพียงพอในการวิเคราะห์การเข้ารหัส THEN
            วิเคราะห์การเข้ารหัสแบบอักษรด้วย CFontRegDlg::DoModal()
        END IF
    } END IF
}
```

TextExtract() ทำการถอดรหัสข้อความในเอกสารพีดีเอฟ โดยจะทำการแยกวัตถุข้อความออกมาจากเอกสารพีดีเอฟแล้วแปลงรหัสตัวอักษรให้ตรงตามข้อกำหนด มอก.620

```
CThaiPDF::TextExtract(PDWord word, int FontEncode)
{
    แยกข้อความออกมาจากเอกสารพีดีเอฟ
    CASE ประเภทการเข้ารหัส
        ประเภทที่ 2 หรือ เข้ารหัสตามมาตรฐานที่เอกสารพีดีเอฟกำหนด
            Return ข้อความ
        ประเภทที่ 3
            บวกรหัสตัวอักษรด้วย 358
        ประเภทที่ 4
            แปลงรหัสตัวอักษรตามชื่อตัวอักษร
        ประเภทที่ 5
            แปลงรหัสตัวอักษรจากรหัสยูนิโค้ดเป็นรหัส มอก.620
    END CASE
    Return ข้อความ
}
```

```

ThaiPatternMatching() ทำการค้นข้อความโดยวิธีการเปรียบเทียบสายอักขระ
CThaiPDF::ThaiPatternMatching(Cstring pattern, Cstring Text, BOOL SubPattern)
{
    WHILE(ไม่สิ้นสุดข้อความ) {
        เริ่มต้นอักขระที่จะเปรียบเทียบที่ตัวอักขระตัวแรกของข้อความ
        IF ใช้ข้อความย่อยของสระอำ และ อักขระข้อความจากเอกสารคือสระอา THEN
            ให้ อักขระข้อความจากเอกสาร เท่ากับ สระอำ
        END IF
        IF อักขระข้อความจากเอกสาร คือ 'อ' และอักขระที่ต้องการค้นคือ สระอำ THEN
            IF อักขระ 'อ' ไม่ใช่อักขระสุดท้าย และ
                อักขระข้อความจากเอกสารถัดไปคือสระอา THEN
                ให้ อักขระข้อความจากเอกสาร เท่ากับ สระอำ
            IF อักขระ 'อ' ใช้อักขระสุดท้าย THEN
                ให้ ข้อความย่อยของสระอำ เป็นจริง
            END IF
        END IF
        IF อักขระของข้อความที่ต้องการค้นตรงกับอักขระข้อความในเอกสาร THEN
            เลื่อนอักขระของข้อความที่ต้องการค้นไปด้านขวา 1 อักขระ
        ELSE
            IF อักขระตัวแรกของข้อความที่ค้นตรงกับอักขระข้อความในเอกสาร THEN
                เลื่อนอักขระของข้อความที่ต้องการค้นไปอยู่อักขระตัวที่สอง
            ELSE
                เลื่อนอักขระของข้อความที่ต้องการค้นไปอยู่อักขระตัวที่สอง
            END IF
        END IF
        IF ใช้ข้อความย่อย THEN Return จำนวนอักขระที่ตรงกัน END IF
    END IF
    IF จำนวนอักขระที่ตรงกัน เท่ากับ ขนาดของข้อความที่ต้องการค้น THEN
        Return จำนวนอักขระที่ตรงกัน
    END IF เลื่อนอักขระเป็นอักขระถัดไปของข้อความจากเอกสาร
    END LOOP
}

```

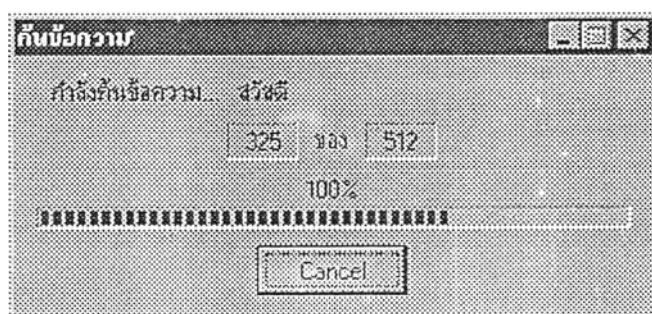
```

Hilite() แสดงสีที่ทับข้อความที่พบ
CThaiPDF::HiliteText(PDDoc CurrentPDDoc, int CurrentPage,
                    PDWord word, int NumWord) {
    เปิดไปยังหน้าเอกสารที่พบข้อความ
    กำหนดให้ตำแหน่งเริ่มต้นในการแสดงสีที่ทับเท่ากับตำแหน่งข้อความที่พบ
    กำหนดให้ขนาดข้อความที่จะแสดงสีที่ทับเท่ากับจำนวนข้อความที่พบ
    แสดงสีที่ทับข้อความที่พบ
}

```

คลาส CProgressDlg

แสดงให้เห็นผู้ใช้ทราบว่าส่วนจำเพาะกำลังค้นข้อความอยู่ ผู้ใช้สามารถที่จะยกเลิกการค้นในขณะที่ยังจำเพาะกำลังค้นเอกสารอยู่ได้ (ดูรูปที่ 25 ประกอบ) ประกอบด้วยฟังก์ชันต่างๆ ดังนี้



รูปที่ 25 หน้าต่างของคลาส CProgressDlg

CProgressDlg() กำหนดค่าเริ่มต้นให้กับคลาสด้วยตัวคลาสเอง

```
CProgressDlg::CProgressDlg(CWnd* pParent /*=NULL*/)
: CDialog(CProgressDlg::IDD, pParent)
{
    กำหนดให้ตัวแปรยกเลิกการทำงานเป็นเท็จ
    กำหนดให้การแสดงหน้าต่างสามารถรับการกดคีย์ได้
}

```

OnInitDialog() กำหนดค่าเริ่มต้นวัตถุของคลาส ให้แสดงหน้าต่างเริ่มต้นในการค้นและ

จำนวนหน้าทั้งหมดในเอกสาร อัตราส่วนของผลการค้นให้เริ่มต้นที่ 0

```
BOOL CProgressDlg::OnInitDialog()
```

```

{
    กำหนดช่วงของการแสดงว่าโปรแกรมกำลังทำงาน ตั้งแต่ 0 ถึง 100
    กำหนดตำแหน่งเริ่มต้นในการแสดงเท่ากับ 0
    แสดงเลขหน้าที่จะเริ่มต้นค้นข้อความ
    แสดงเลขหน้าสุดท้ายของเอกสาร
}

```

OnCancel() ผู้ใช้ยกเลิกการค้นข้อความ

```
Void CprogressDlg::OnCancel()
```

```

{
    กำหนดให้ตัวแปรยกเลิกการทำงานมีค่าเป็นจริง แล้วปิดหน้าต่างวัตถุของคลาสนี้
}

```

PeekAndPump() อนุญาตให้ขณะที่โปรแกรมกำลังค้นข้อความ หน้าต่างของวัตถุสามารถที่จะรับการกดคีย์ของผู้ใช้ได้

```
void CProgressDlg::PeekAndPump(BOOL bCancelOnESCkey /*= TRUE*/)

```

```

{
    IF มีการแสดงหน้าต่างอื่นที่ไม่ใช่หน้าต่างของวัตถุในคลาสนี้ THEN
        ให้แสดงหน้าต่างของวัตถุในคลาสนี้
    END IF
    WHILE ไม่มีการยกเลิกจากผู้ใช้
    {
        IF ตรวจสอบว่ามีการกดคีย์ใดๆหรือไม่ และ คีย์นั้นเท่ากับยกเลิก THEN
            ยกเลิกการทำงานและปิดหน้าต่างวัตถุของคลาสนี้
        END IF
        แสดงหน้าต่างและทำงานต่อไป
    }
}

```

คลาส CFontRegDlg

จะทำหน้าที่ช่วยในการวิเคราะห์การเข้ารหัสแบบอักษรในเอกสาร ในกรณีที่มีข้อมูลในเอกสารไม่เพียงพอต่อการวิเคราะห์การเข้ารหัส (ดูรูปที่ 26 ประกอบ) มีฟังก์ชันต่างๆ ดังนี้



```
Void CFontRegDlg::OnRegFontType2() {
    กำหนดให้ประเภทการเข้ารหัสของแบบอักษรเท่ากับ 2
}
```

OnRegFontType3() เมื่อมีการเลือกตัวเลือกนี้ให้ตัวแปรประเภทการเข้ารหัสเท่ากับ 3 (คือ รหัสตัวอักษรลดลงจากข้อกำหนด มอก.620 ด้วยค่าคงที่ 35<sub>๕</sub>)

```
Void CFontRegDlg::OnRegFontType3() {
    กำหนดให้ประเภทการเข้ารหัสของแบบอักษรเท่ากับ 3
}
```

DoDataExchange() จะนำค่าที่ได้รับในองค์ประกอบของหน้าต่างคลาสนี้ มาเก็บไว้ในตัวแปรที่กำหนด เช่น ชื่อแบบอักษร ข้อความที่เป็นตัวเลือก

```
void CFontRegDlg::DoDataExchange(CDataExchange* pDX) {
    แสดงชื่อแบบอักษรที่จะทำการวิเคราะห์
    แสดงข้อความที่เป็นตัวเลือก
}
```

OnOK() ปิดหน้าต่างวัตถุเมื่อมีการกดปุ่มตกลง แล้วทำการส่งค่าตัวเลือกให้กับตัวแปรที่กำหนด

```
void CFontRegDlg::OnOK() {
    ปิดหน้าต่างวัตถุของคลาสนี้
    เก็บค่าตัวเลือกไว้ที่ตัวแปรประเภทการเข้ารหัสแบบอักษร
}
```

OnCancel() ปิดหน้าต่างวัตถุแล้วกำหนดให้ตัวแปรประเภทการเข้ารหัสแบบอักษรเท่ากับ

1

```
void CFontRegDlg::OnCancel() {
    ปิดหน้าต่างวัตถุของคลาสนี้
    กำหนดให้ตัวแปรประเภทการเข้ารหัสแบบอักษรเท่ากับ 1
}
```