



## รายงาน

### เรื่อง โปรแกรม Borturn Library

## จัดทำโดย

นางสาวกฤษฎา

โมรา

รหัสนักศึกษา 6304062630016

นางสาวณัชชา

วิเศษสุทธิ

รหัสนักศึกษา 6304062630091

## เสนอ

อาจารย์กอบเกียรติ สระอุบล

รายงานเล่มนี้เป็นส่วนหนึ่งของวิชา

Computer Programming I รหัสวิชา 040613102

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

ภาคเรียนที่ 1 ปีการศึกษา 2563

## คำนำ

การจัดทำรายงาน เรื่อง “โปรแกรม Borturn Library” นี้เป็นส่วนหนึ่งของวิชา Computer Programming I ของหลักสูตรวิทยาศาสตรบัณฑิต ภาควิชาวิทยาการคอมพิวเตอร์และสารสนเทศ คณะวิทยาศาสตร์ประยุกต์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ สำหรับนักศึกษาที่กำลังศึกษาในชั้นปีที่หนึ่งเพื่อให้นักศึกษาได้นำความรู้ที่เรียนมาประยุกต์ใช้ในการเขียนและพัฒนาโปรแกรม

“โปรแกรม Borturn Library” จัดทำขึ้นเพื่ออำนวยความสะดวกให้แก่ผู้ใช้งานในห้องสมุด โดยผู้ใช้งานสามารถเข้าถึงห้องสมุดได้สะดวกขึ้นโดยจะสามารถตรวจสอบหนังสือทั้งหมดที่อยู่ในห้องสมุด สถานะของหนังสือแต่ละเล่ม การยืม-คืนหนังสือ และการชำระค่าปรับเมื่อยืมหนังสือเกินกำหนดเวลา

คณะผู้จัดทำหวังว่า “โปรแกรม Borturn Library” จะอำนวยความสะดวกและเป็นประโยชน์ต่อผู้ใช้งาน หากผิดพลาดประการใด คณะผู้จัดทำต้องขออภัยมา ณ ที่นี้ด้วย

คณะผู้จัดทำ

## สารบัญ

เรื่อง	หน้า
1. หลักการทำงานของโปรแกรม	1
2. User Interface	1
3. Input	4
4. Output	4
5. Flowchart การทำงานรวม	5
6. Flowchart แสดงการทำงานของส่วนหลัก	6
7. ข้อมูล Data file	11
8. โครงสร้าง List	12
9. Code	13

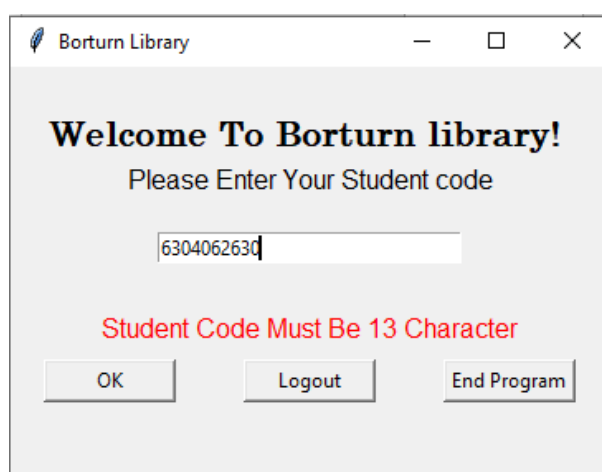
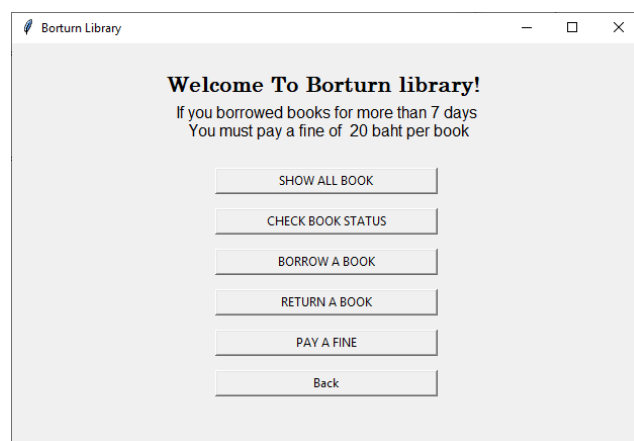
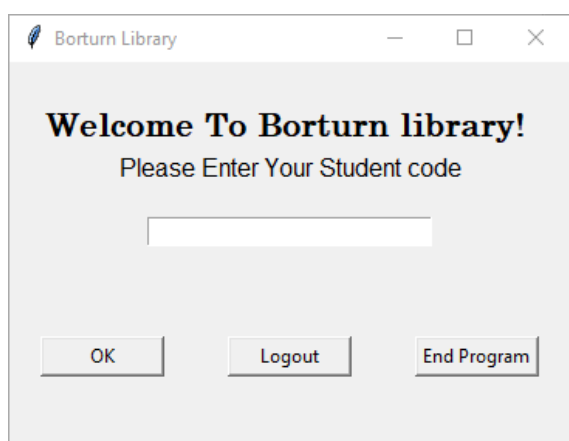
## หลักการการทำงานของโปรแกรม

โปรแกรม Borturn Library เป็นโปรแกรมสำหรับช่วยอำนวยความสะดวกในการใช้งานห้องสมุด โดยห้องสมุดนี้จะมีหนังสือรวมทั้งสิ้น 30 เล่ม สามารถยืมหนังสือ คืนหนังสือ แสดงสถานะของหนังสือว่าหนังสือเล่มนั้นว่างอยู่หรือไม่ และมีระบบชำระเงินค่าปรับหากยืมหนังสือนานเกิน 7 วัน โดยปรับเล่มละ 20 บาท ตัวโปรแกรมจะมีข้อความว่า 'Welcome To Borturn library!' บรรทัดถัดไปจะบอกถึงเงื่อนไขของการจ่ายค่าปรับ และจะมีเมนู 6 เมนูด้วยกันให้ผู้ใช้เลือกใช้งาน

## User Interface

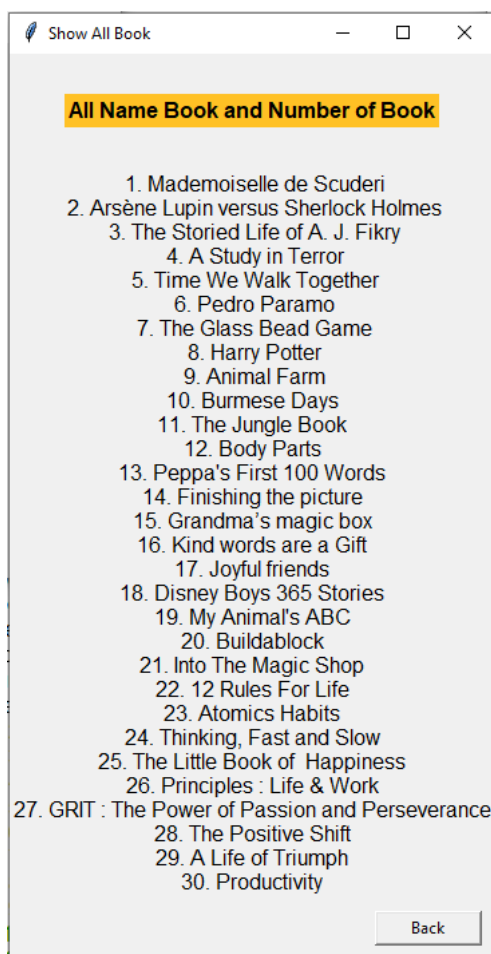
### 1. หน้าแรกของโปรแกรม

เมื่อเปิดโปรแกรมจะปรากฏหน้าจอให้ผู้ใส่กรอกรหัสนักศึกษา เมื่อปุ่ม OK จะแสดงหน้าต่างของเมนู หากกรอกตัวเลขไม่ครบหรือเกิน 13 ตัว หรือไม่กรอกตัวเลข โปรแกรมจะไม่ปรากฏหน้าต่างของเมนูแต่จะปรากฏข้อความว่า "Student Code Must Be 13 Character" และเมื่อผู้ใช้ทำการเรียบร้อยแล้วหรือพบว่ากรอกรหัสผิดให้ทำการกดปุ่ม Logout เท่านั้น ห้ามลบตัวเลขออกทางคีย์บอร์ด และเมื่อกดปุ่ม End Program โปรแกรมจะสรุปจำนวนหนังสือที่ถูกยืมหรือถูกคืนและจบการทำงานของโปรแกรม



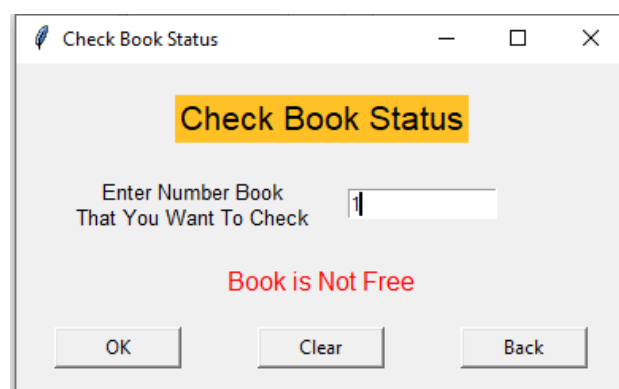
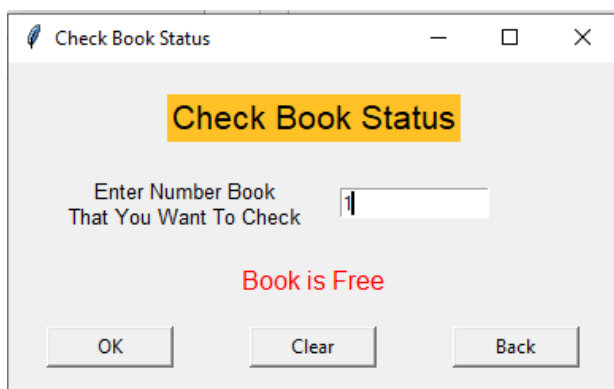
## 2. Show All Book

เมื่อกดเมนู SHOW ALL BOOK จะปรากฏหน้าต่างแสดงรหัสและชื่อของหนังสือทั้งหมด จำนวนทั้งสิ้น 30 เล่ม



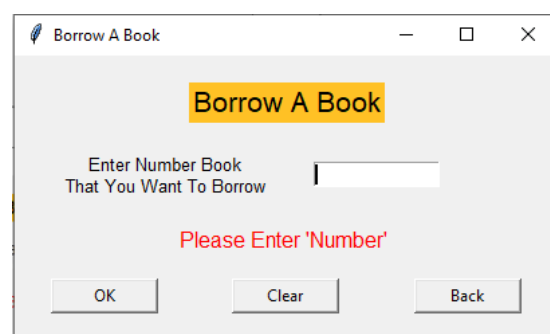
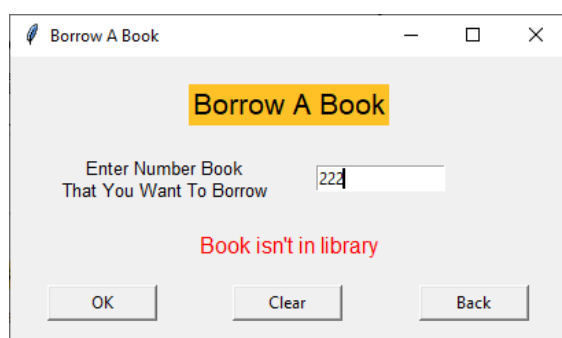
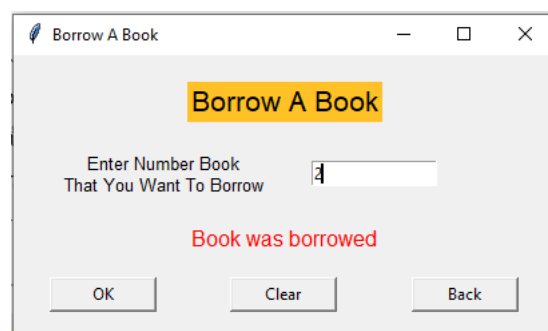
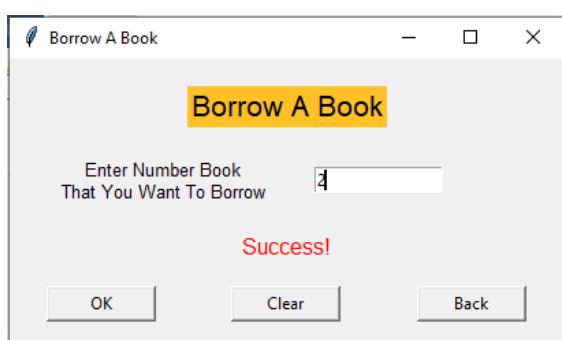
## 3. Check Book Status

เมื่อกดปุ่ม CHECK BOOK STATUS จะปรากฏหน้าต่างให้ผู้ใส่กรอกรหัสของหนังสือที่ต้องการจะตรวจสอบสถานะของหนังสือ เมื่อกดปุ่ม OK หากยังไม่ถูกยืมจะแสดงข้อความว่า “Book is Free” หากถูกยืมไปแล้วจะแสดงข้อความว่า “Book is Not Free” ปุ่ม Clear จะทำการลบตัวเลขให้โดยไม่ต้องกดคีย์บอร์ด



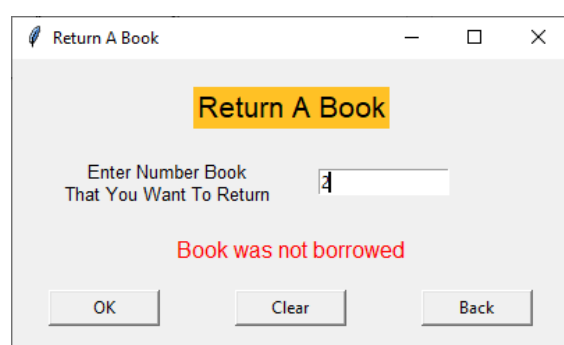
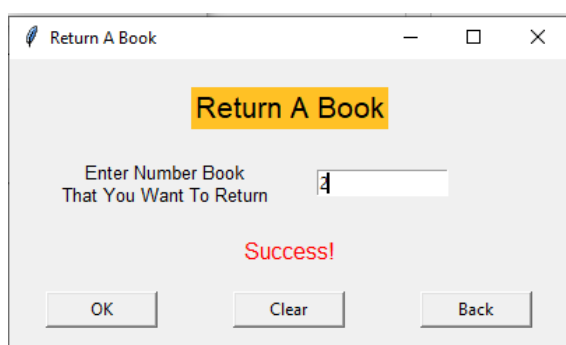
#### 4. Borrow A Book

เมื่อกดปุ่ม BORROW A BOOK จะปรากฏหน้าต่างให้ผู้ใช้อกรรหัสของหนังสือที่ต้องการจะยืมเมื่อกดปุ่ม OK ถ้าหนังสือเล่มนั้นว่างอยู่จะแสดงข้อความว่า “Success!” หากหนังสือเล่มนั้นถูกยืมไปแล้วจะแสดงข้อความว่า “Book was borrowed” หากกรอกตัวเลขที่เกิน 30 หรือกรอกตัวเลขที่น้อยกว่า 1 จะแสดงข้อความว่า “Book isn't in library” และหากไม่กรอกตัวเลขแต่กดปุ่ม OK จะแสดงข้อความว่า “Please Enter 'Number'”



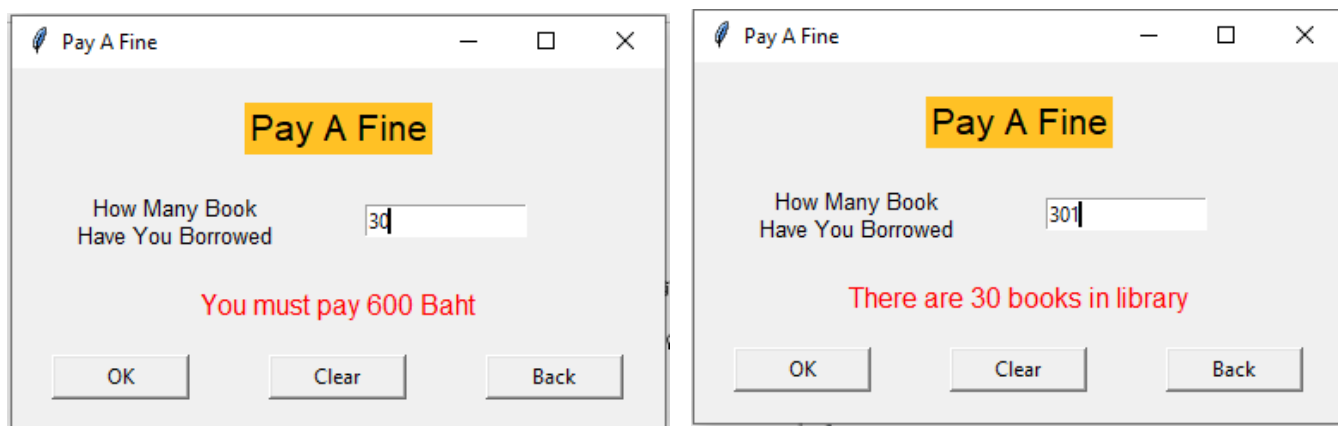
#### 5. Return A Book

เมื่อกดปุ่ม BORROW A BOOK จะปรากฏหน้าต่างให้ผู้ใช้อกรรหัสของหนังสือที่ต้องการจะคืนเมื่อกดปุ่ม OK ถ้าหนังสือเล่มถูกยืมไปแล้วอยู่จะแสดงข้อความว่า “Success!” หากหนังสือเล่มนั้นยังไม่ได้ถูกยืมไปจะแสดงข้อความว่า “Book was not borrowed” หากกรอกตัวเลขที่เกิน 30 หรือกรอกตัวเลขที่น้อยกว่า 1 จะแสดงข้อความว่า “Book isn't in library” และหากไม่กรอกตัวเลขแต่กดปุ่ม OK จะแสดงข้อความว่า "Please Enter 'Number'"



## 6. Pay A Fine

เมื่อกดปุ่ม PAY A FINE จะปรากฏหน้าต่างให้ผู้กรอกจำนวนของหนังสือที่ต้องการถูกยืมไปเมื่อกดปุ่ม OK จะแสดงจำนวนเงินที่ต้องจ่าย หากกรอกตัวเลขที่เกิน 30 หรือกรอกตัวเลขที่น้อยกว่า 1 จะแสดงข้อความว่า “There are 30 books in library” หากไม่กรอกตัวเลขแต่กดปุ่ม OK จะแสดงข้อความว่า “Please Enter 'Number'”

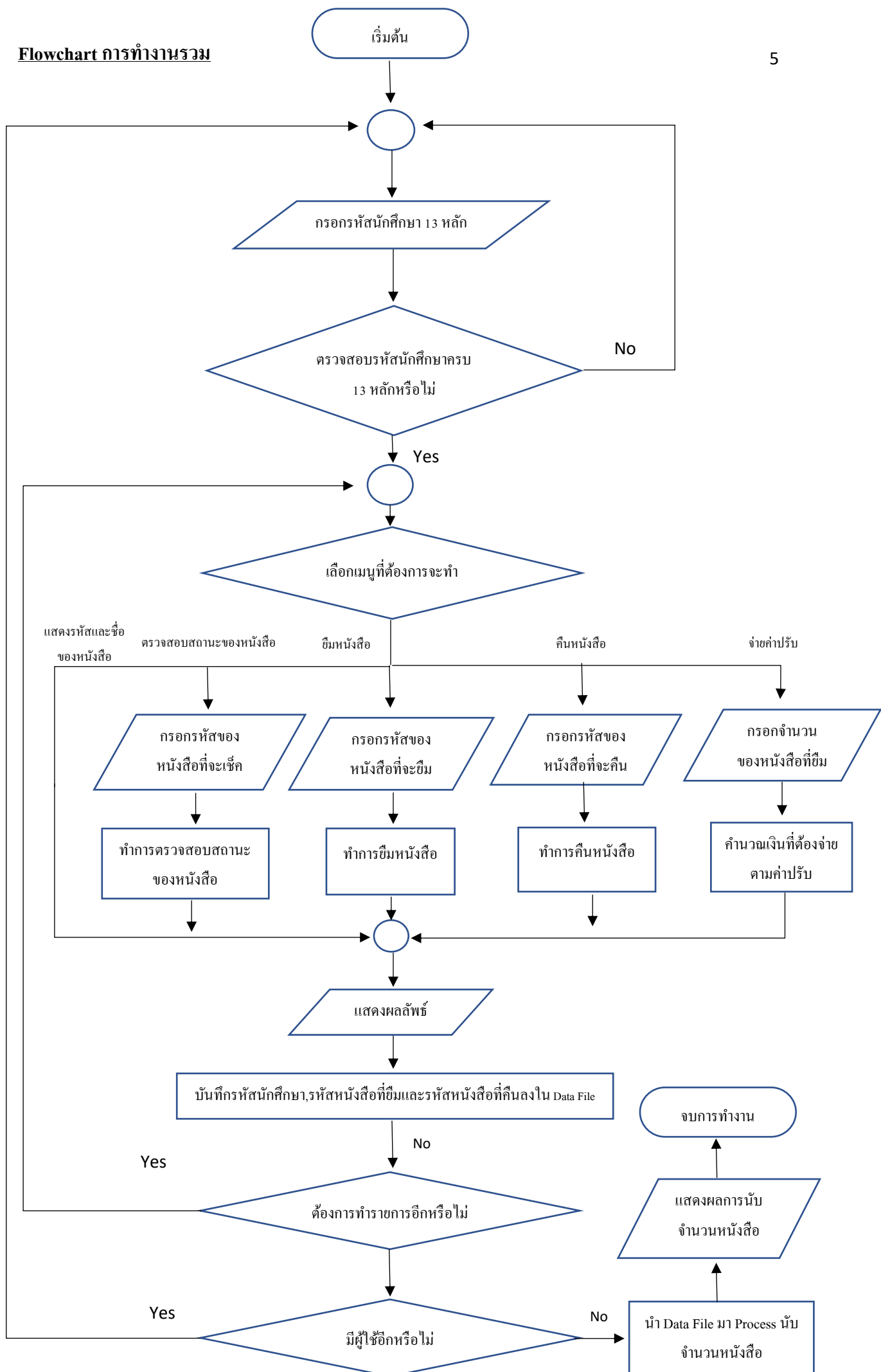


### Input

1. รหัสนักศึกษา 13 หลัก
2. รหัสหนังสือที่ต้องการจะเช็คสถานะ
3. รหัสหนังสือที่ต้องการจะยืม
4. รหัสหนังสือที่ต้องการจะคืน
5. จำนวนหนังสือที่ยืมเกิน 7 วัน

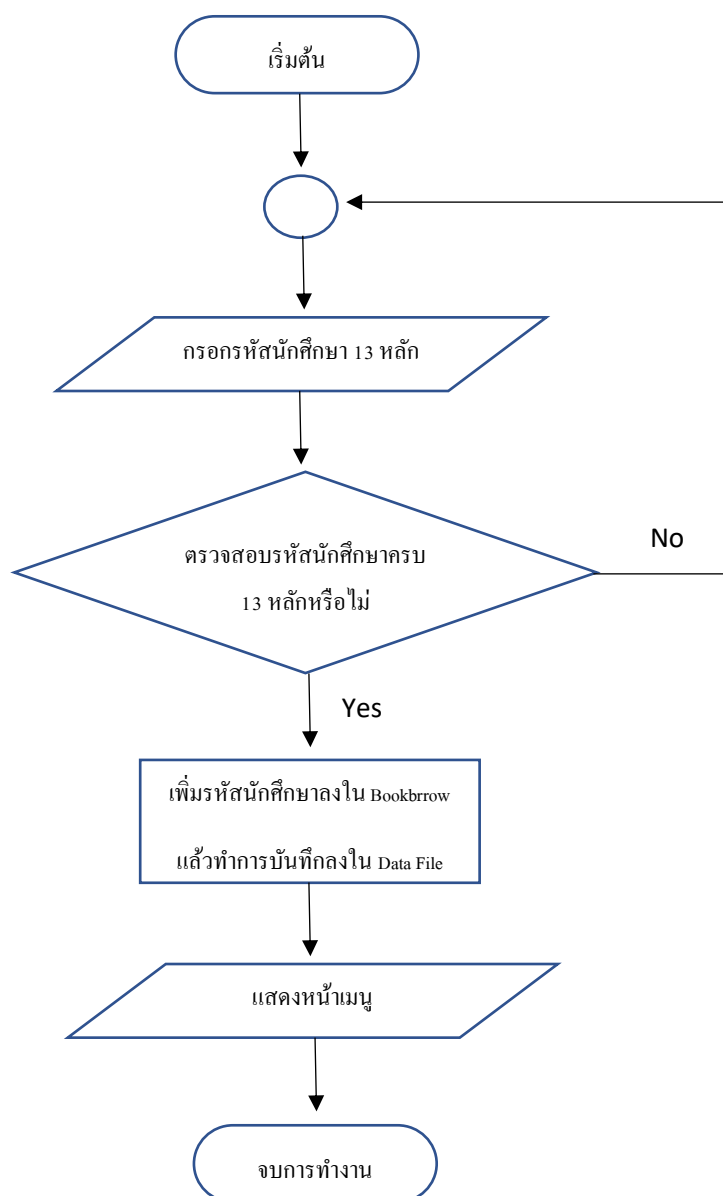
### Output

1. สถานะของหนังสือที่เช็ค
2. การยืมหนังสือสำเร็จหรือไม่
3. การคืนหนังสือสำเร็จหรือไม่
4. จำนวนเงินที่ต้องจ่ายทั้งหมด

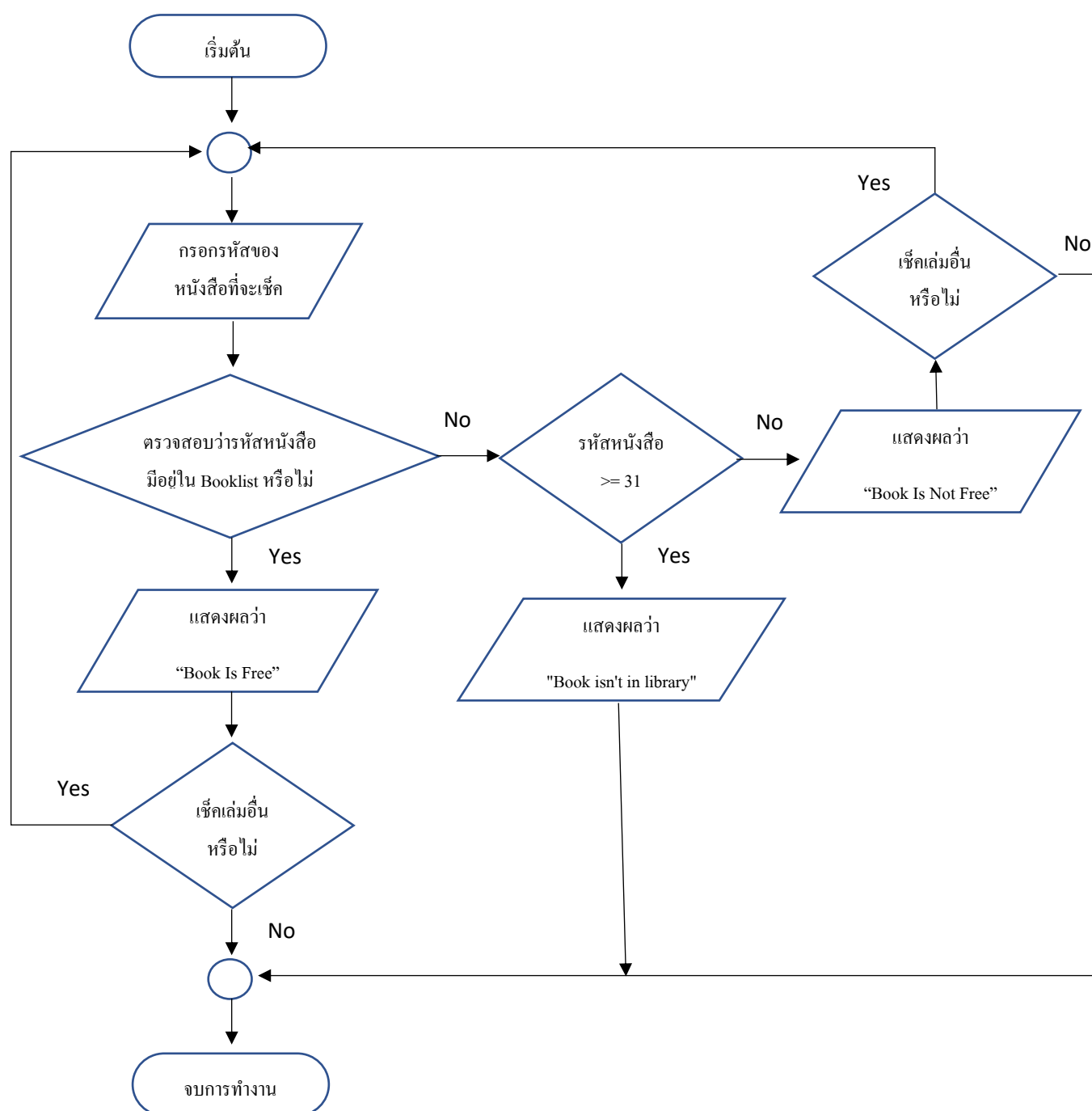




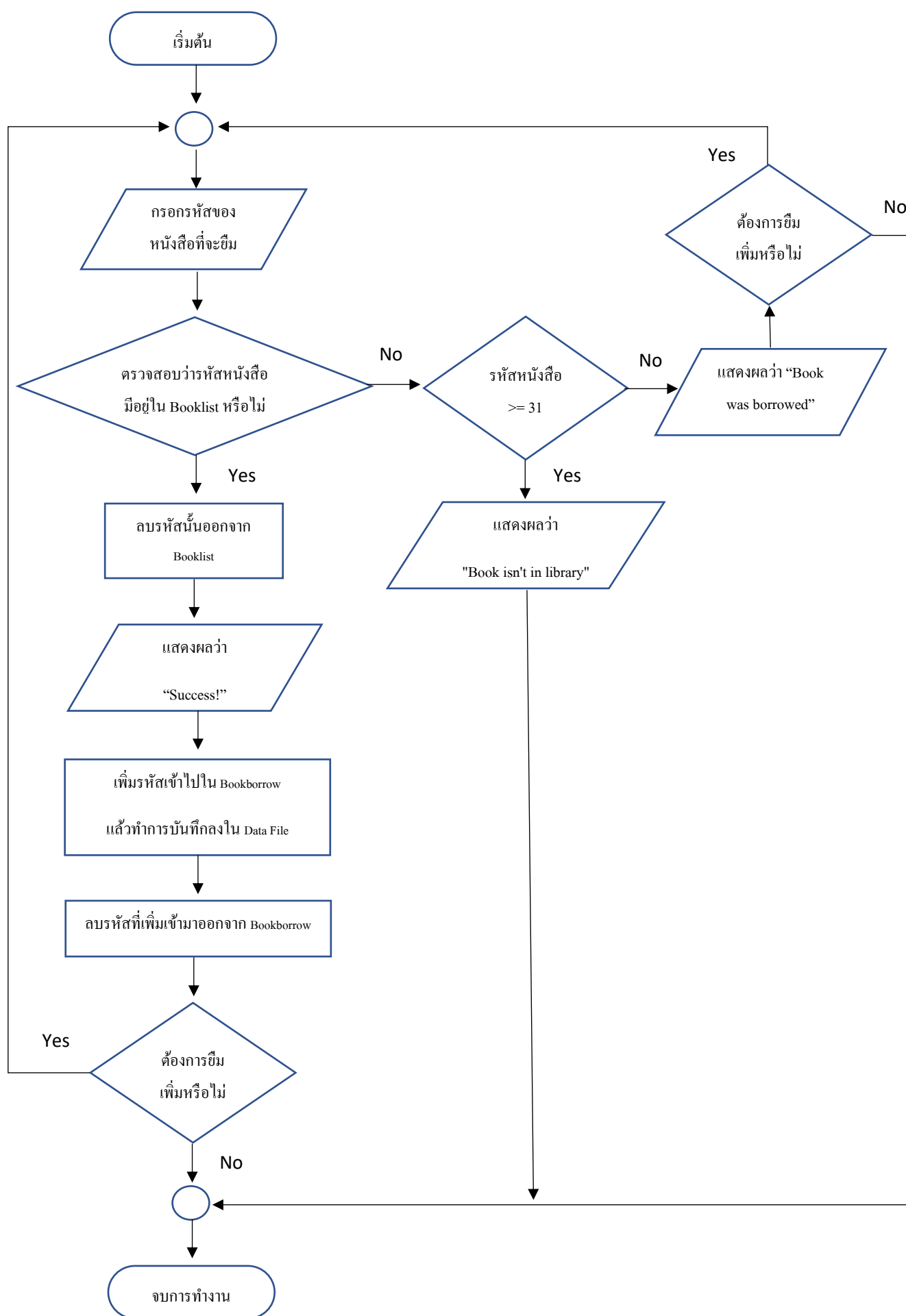
### Flowchart แสดงการทำงานส่วนหลักของส่วนรับค่ารหัสนักศึกษา



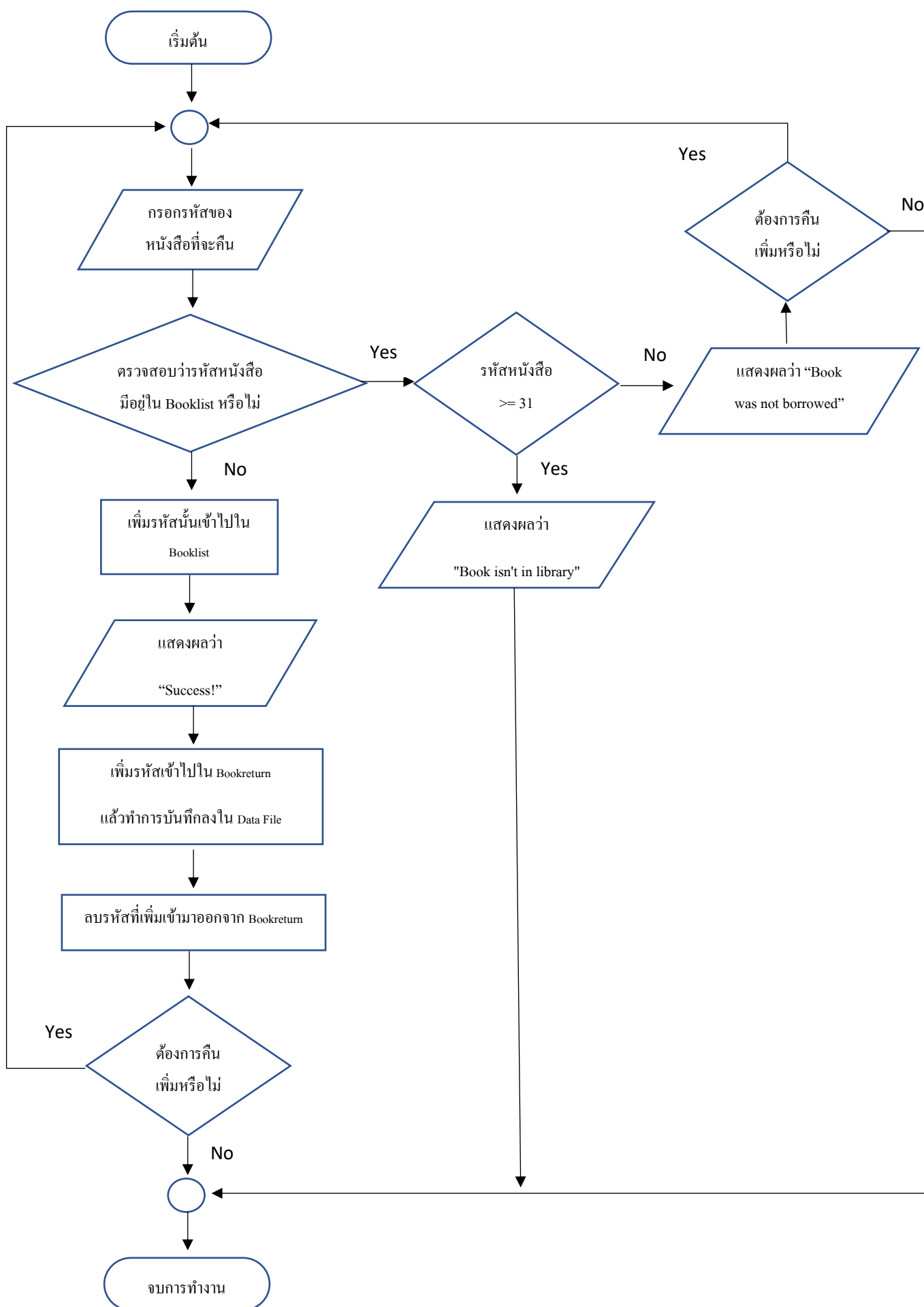
# Flowchart แสดงการทำงานส่วนหลักของส่วนเช็คสถานะหนังสือ



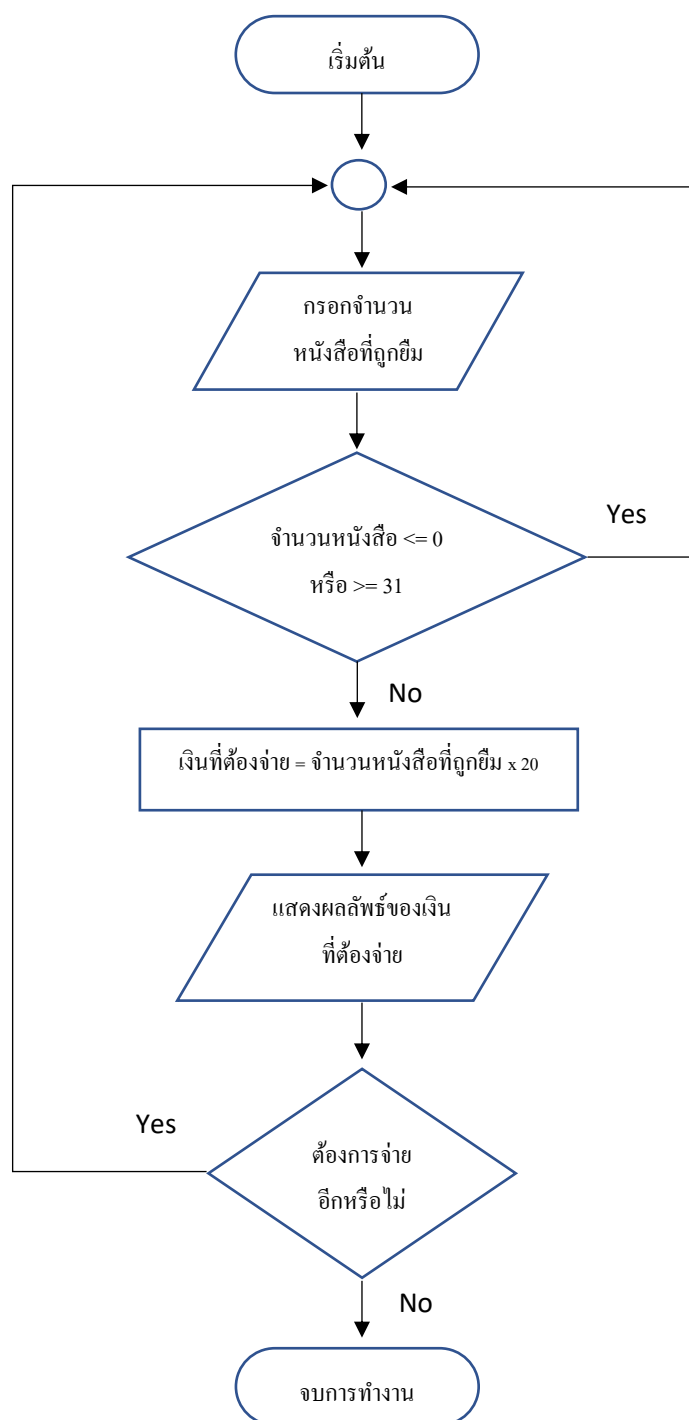
# Flowchart แสดงการทำงานของส่วนยืมหนังสือ



### Flowchart แสดงการทำงานส่วนของส่วนคืนหนังสือ



### Flowchart แสดงการทำงานของส่วนจ่ายเงินค่าปรับ



5.



## โครงสร้าง List

Booklist จะเก็บข้อมูลของรหัสหนังสือทั้งหมดตั้งแต่ 1- 30 ถ้ายืมหนังสือจะใช้เมธอด remove(x: object) ลบตัวเลขที่ Input เข้ามาออกจาก list และถ้าต้องการคืนหนังสือจะใช้เมธอด append(x: object) เพิ่มตัวเลขที่ Input ใส่เข้าไปใน list

Bookborrow และ Bookreturn จะใช้สำหรับเก็บข้อมูลของรหัสนักศึกษา รหัสของหนังสือที่ถูกยืม และรหัสของหนังสือที่ถูกคืนเพื่อบันทึกลงใน Data File เมื่อทำการกรอกรหัสนักศึกษารหัสนั้นจะถูกเก็บไว้ในตัวลิสต์ และเมื่อมีการยืมหนังสือหรือคืนหนังสือรหัสนั้นก็จะถูกเก็บไว้ในลิสต์โดยใช้เมธอด append(x:object) เช่นเดียวกัน และบันทึกลงใน Data File ถ้านักศึกษาคนเดิมต้องการยืมเพิ่มก็จะใช้ เมธอด remove(x: object) จากนั้นจะ input รหัสเข้าไปใหม่แล้วทำแบบเดิม หากเปลี่ยนรหัสนักศึกษาจะใช้ เมธอด clear() เพื่อลบข้อมูลลิสต์อันเก่าทิ้งให้หมด และรับค่าใหม่อีกครั้ง

```
booklist = [num for num in range(1,31)]

bookborrow = []
bookreturn = []
```

### Code

```

from tkinter import *
import csv

booklist = [num for num in range(1,31)]

bookborrow = []
bookreturn = []

def write(): #บันทึกข้อมูลการยืม
    filepath = 'borrow.csv'
    with open(filepath,"a",encoding='utf-8') as infile:
        writer = csv.writer(infile,lineterminator = "\n")
        writer.writerow(bookborrow)

def write1(): #บันทึกข้อมูลการคืน
    filepath = 'return.csv'
    with open(filepath,"a",encoding='utf-8') as infile:
        writer = csv.writer(infile,lineterminator = "\n")
        writer.writerow(bookreturn)

def namebook(): #แสดงชื่อหนังสือกับรหัสหนังสือทั้งหมด
    shownamebook1 = Tk()
    shownamebook1.title("Show All Book")
    shownamebook1.minsize(300,280)
    head = Label(shownamebook1,text = "All Name Book and Number of Book",font = "Arial 12 bold")
    head.grid(row=0,columnspan=4,pady=30)
    head.configure(bg='goldenrod1')
    book = Label(shownamebook1 ,text = " 1. Mademoiselle de Scuder\n"
        + " 2. Arsène Lupin versus Sherlock Holmes\n"
        + " 3. The Storied Life of A. J. Fikry\n"
        + " 4. A Study in Terror\n"
        + " 5. Time We Walk Together\n"
        + " 6. Pedro Paramo\n"
        + " 7. The Glass Bead Game\n"
        + " 8. Harry Potter\n"
        + " 9. Animal Farm\n"
        + "10. Burmese Days\n"

```



```

+"11. The Jungle Book\n"
+"12. Body Parts\n"
+"13. Peppa's First 100 Words\n"
+"14. Finishing the picture\n"
+"15. Grandma's magic box\n"
+"16. Kind words are a Gift\n"
+"17. Joyful friends\n"
+"18. Disney Boys 365 Stories\n"
+"19. My Animal's ABC\n"
+"20. Buildablock\n"
+"21. Into The Magic Shop\n"
+"22. 12 Rules For Life\n"
+"23. Atomics Habits\n"
+"24. Thinking, Fast and Slow\n"
+"25. The Little Book of Happiness\n"
+"26. Principles : Life & Work\n"
+"27. GRIT : The Power of Passion and Perseverance\n"
+"28. The Positive Shift\n"
+"29. A Life of Triumph\n"
+"30. Productivity"
,font = "Arial 12")

book.grid(row=1,column=1)

btClosename = Button(shownamebook1,text="Back",width=10,command=shownamebook1.destroy)
btClosename.grid(row=5,column=1,pady=10,sticky=SE,padx=10)

def winstatus(): #ตรวจสอบสถานะของหนังสือ
    def statusbook1():
        try:
            num = int(inp.get())
            if num in booklist:
                lbl["text"] = "Book is Free"
            elif num not in booklist:
                if 1<= num <=30:
                    lbl["text"] = "Book is Not Free"
                elif num <=0 or num >=31:
                    lbl["text"] = "Book isn't in library"

```

---

```

except Exception as e:
    lbl["text"] ="Please Enter 'Number' "

def clear():
    inp.delete(0,"end")

checkstatus1 = Tk()
checkstatus1.title("Check Book Status")
checkstatus1.minsize(100,210)

lbl = Label(checkstatus1,pady=8,text="",fg="red",font="Arial 12")
lbl.grid(row=3,column=0,pady=10,columnspan=3)

head = Label(checkstatus1,text = "Check Book Status",font = "Arial 16")
head.grid(row=1,column=0,columnspan=3,padx=100,pady=20)
head.configure(bg='goldenrod1')

head1 = Label(checkstatus1,text = "Enter Number Book\nThat You Want To Check",font = "Arial 10")
head1.grid(row=2,column=0,columnspan=2,sticky=W,padx=35)

inp = Entry(checkstatus1,width=15)
inp.grid(row=2,column=1,columnspan=3)
inp.focus()

btOK= Button(checkstatus1,text="OK",width=10,command=statusbook1)
btOK.grid(row=4,column=0)

btCLEA = Button(checkstatus1,text = 'Clear',width = 10,command = clear)
btCLEA.grid(row=4,column=1)

bt = Button(checkstatus1,text="Back",command = checkstatus1.destroy,width=10)
bt.grid(row=4,column=2)

```

```

def borrowbook(): #ทำการยืมหนังสือ
    def borrowbook1():
        try:
            num = int(inp.get())
            if num in booklist:
                booklist.remove(num)
                bookborrow.append(num)
                write()
                bookborrow.remove(num)

                lbl["text"] = "Success!"
            elif num not in booklist:
                if num <=0 or num >=31:
                    lbl["text"] = "Book isn't in library"
                elif 1 <= num <= 30:
                    lbl["text"] = "Book was borrowed"
            except Exception as e:
                lbl["text"] = "Please Enter 'Number' "

    def clear():
        inp.delete(0,"end")

    borrow1 = Tk()
    borrow1.title("Borrow A Book")
    borrow1.minsize(100,210)

    head = Label(borrow1,text = "Borrow A Book",font = "Arial 16")
    head.grid(row=1,column=0,columnspan=3,pady=20,padx=130)
    head.configure(bg='goldenrod1')

    head1 = Label(borrow1,text = "Enter Number Book\nThat You Want To Borrow",font = "Arial 10")
    head1.grid(row=2,column=0,columnspan=2,sticky=W,padx=35)

    inp = Entry(borrow1,width=15)
    inp.grid(row=2,column=1,columnspan=3)
    inp.focus()

```

```

lbl = Label(borrow1,pady=8,text="",fg="red",font="Arial 12")
lbl.grid(row=3,column=0,pady=10,columnspan=3)

btOK= Button(borrow1,text="OK",width=10,command=borrowbook1)
btOK.grid(row=4,column=0)

btCLEAR = Button(borrow1,text= 'Clear',width = 10,command = clear)
btCLEAR.grid(row=4,column=1)

bt = Button(borrow1,text="Back",command = borrow1.destroy,width=10)
bt.grid(row=4,column=2)

```

```

def returnbook(): #ทำการคืนหนังสือ
    def returnbook1():
        try:
            num = int(inp.get())
            if num not in booklist:
                if num <=0 or num >= 31:
                    lbl["text"] = "Book isn't in library"
                elif 1 <= num <= 30:
                    booklist.append(num)
                    bookreturn.append(num)
                    write1()
                    bookreturn.remove(num)
                    lbl["text"] = "Success!"
            elif num in booklist:
                if 1 <= num <= 30:
                    lbl["text"] = "Book was not borrowed"
            except Exception as e:
                lbl["text"] = "Please Enter 'Number' "

    def clear():
        inp.delete(0,"end")

return1 = Tk()
return1.title("Return A Book")
return1.minsize(100,210)

```

```

head = Label(return1,text = "Return A Book",font = "Arial 16")
head.grid(row=1,column=0,columnspan=3,pady=20,padx=130)
head.configure(bg='goldenrod1')

head1 = Label(return1,text = "Enter Number Book\nThat You Want To Return",font = "Arial 10")
head1.grid(row=2,column=0,columnspan=2,sticky=W,padx=35)

inp = Entry(return1,width=15)
inp.grid(row=2,column=1,columnspan=3)
inp.focus()

lbl = Label(return1,pady=8,text="",fg="red",font="Arial 12")
lbl.grid(row=3,column=0,pady=10,columnspan=3)

btOK= Button(return1,text="OK",width=10,command=returnbook1)
btOK.grid(row=4,column=0)

btCLEAR = Button(return1,text = 'Clear',width = 10,command = clear)
btCLEAR.grid(row=4,column=1)

bt = Button(return1,text="Back",command = return1.destroy,width=10)
bt.grid(row=4,column=2)

def pay(): #จ่ายค่าปรับ
    def pay1(baht):
        payfine = baht * 20
        return payfine

    def pay2():
        try:
            baht = int(inp.get())
            money = pay1(baht)
            if 1 <= baht <=30:
                lbl["text"] = "You must pay {} Baht".format(pay1(baht))
            elif baht <=0 or baht >30:
                lbl["text"] = "There are 30 books in library"

```

```

except Exception as e:
    lbl["text"] = "Please Enter 'Number' "

def clear():
    inp.delete(0,"end")

pay = Tk()
pay.title("Pay A Fine")
pay.minsize(200,210)

head = Label(pay,text = "Pay A Fine",font = "Arial 16")
head.grid(row=1,column=0,columnspan=3,pady=20,padx=135)
head.configure(bg='goldenrod1')

head1 = Label(pay,text = "How Many Book\nHave You Borrowed",font = "Arial 10")
head1.grid(row=3,column=0,columnspan=2,sticky=W,padx=35)

inp = Entry(pay,width=15)
inp.grid(row=3,column=1,columnspan=3)
inp.focus()

lbl = Label(pay,pady=8,text="",fg="red",font="Arial 12")
lbl.grid(row=4,column=0,pady=10,columnspan=3)

btOK= Button(pay,text="OK",width=10,command=pay2)
btOK.grid(row=5,column=0)

btCLEAR = Button(pay,text = 'Clear',width = 10,command = clear)
btCLEAR.grid(row=5,column=1)

bt = Button(pay,text="Back",command = pay.destroy,width=10)
bt.grid(row=5,column=2)

```

```

def mainprocess(): #เป็นหน้าเมนูเชื่อมไปยังฟังก์ชันต่าง ๆ
    code = str(inp.get())
    readcode = len(code)
    if readcode != 13:
        lbl["text"] = "Student Code Must Be 13 Character"

    else:
        bookborrow.append(code)
        bookreturn.append(code)
        mainwin = Tk()
        mainwin.title("Borturn Library")
        mainwin.minsize(500,400)

        head = Label(mainwin ,text = "\nWelcome To Borturn library! ",font = "Century 16 bold")
        head.grid(row=0,column=0,columnspan=3,padx=150)

        head1 = Label(mainwin ,text = "If you borrowed books for more than 7 days"
            +"\n You must pay a fine of 20 baht per book\n",font = "Arial 12")
        head1.grid(row=1,column=0,columnspan=3)

        btShow = Button(mainwin,text="SHOW ALL BOOK",width=30,command=namebook)
        btShow.grid(row=2,pady=7,columnspan=3)

        btCheck = Button(mainwin,text="CHECK BOOK STATUS",width=30,command=winstatus)
        btCheck.grid(row=3,pady=7,columnspan=3)

        btBorrow = Button(mainwin,text="BORROW A BOOK",width=30,command=borrowbook)
        btBorrow.grid(row=4,pady=7,columnspan=3)

        btReturn = Button(mainwin,text="RETURN A BOOK",width=30,command=returnbook)
        btReturn.grid(row=5,pady=7,columnspan=3)

        btPay= Button(mainwin,text="PAY A FINE",width=30,command=pay)
        btPay.grid(row=6,pady=7,columnspan=3)

        btEnd= Button(mainwin,text="Back",width=30,command=mainwin.destroy)
        btEnd.grid(row=7,pady=7,columnspan=3)

```

#หน้าต่างแรก รับรหัสนักศึกษา

```
def clear():
```

```
    inp.delete(0,"end")
```

```
    bookborrow.clear()
```

```
    bookreturn.clear()
```

```
userwin = Tk()
```

```
userwin.title("Borturn Library")
```

```
userwin.minsize(300,250)
```

```
head = Label(userwin ,text = "\nWelcome To Borturn library! ",font = "Century 16 bold")
```

```
head.grid(row=0,column=0,columnspan=3,padx=20)
```

```
head1 = Label(userwin ,text = "Please Enter Your Student code",font = "Arial 12")
```

```
head1.grid(row=1,column=0,columnspan=3)
```

```
inp = Entry(userwin,width=30)
```

```
inp.grid(row=2,column=0,columnspan=3,pady = 20)
```

```
inp.focus()
```

```
lbl = Label(userwin,pady=8,text="",fg="red",font="Arial 12")
```

```
lbl.grid(row=3,column=0,columnspan=3)
```

```
btOK= Button(userwin,text="OK",width=10,command=mainprocess)
```

```
btOK.grid(row=4,column=0)
```

```
btlogout = Button(userwin,text = 'Logout',width = 10,command = clear)
```

```
btlogout.grid(row=4,column=1)
```

```
bt = Button(userwin,text="End Program",command = userwin.destroy,width=10)
```

```
bt.grid(row=4,column=2)
```

```
mainloop()
```



```
try:
    filepath = 'borrow.csv'
    with open(filepath,"r",encoding='utf-8') as infile:
        readborrow = csv.reader(infile)
        myborrow = list(readborrow)
        bookborrow = len(myborrow)
        print("All Book Was Borrowed = {}".format(bookborrow))
except Exception as e:
    print("You doesn't borrow book")

try:
    filepath = 'return.csv'
    with open(filepath,"r",encoding='utf-8') as infile:
        readreturn = csv.reader(infile)
        myreturn = list(readreturn)
        bookreturn = len(myreturn)
        print("All Book Was Returned = {}".format(bookreturn))
except Exception as e:
    print("You doesn't return book")
```