

แนวคำตอบใบกิจกรรมที่ 8.1

ค่าโดยสาร

1. การวิเคราะห์และกำหนดรายละเอียดของปัญหา

- 1) ข้อมูลเข้า คือจำนวนคนขึ้นรถทั้งหมด จำนวนผู้สูงอายุ และจำนวนเด็ก
- 2) ข้อมูลออก คือราคาค่าโดยสารหลังหักส่วนลดแล้ว.....
- 3) วิธีการตรวจสอบความถูกต้อง โดยสมมติข้อมูลทดสอบดังนี้

ผู้โดยสารทั้งหมด	ต่ำกว่า 3 ขวบ	สูงกว่า 60 ปี	ค่าโดยสารทั้งหมด
20	3	3	155
40	5	6	320
15	4	2	100

2. การวางแผนการแก้ปัญหา

ดูได้จากหนังสือเรียนบทที่ 2 การแก้ปัญหาด้วยไพทอน หัวข้อ 2.1

3. การดำเนินการแก้ปัญหา

```
all = int(input('มีผู้โดยสารทั้งหมดกี่คน: '))
child = int(input('มีผู้โดยสารอายุต่ำกว่า 3 ขวบกี่คน: '))
elders = int(input('มีผู้โดยสารอายุตั้งแต่ 60 ปีขึ้นไปกี่คน : '))

def fare_elders(a):
    return a*(10/2)

def fare_regular(a):
    return a*10

regular = all-child-elders
total = fare_regular(regular)+fare_elders(elders)
if all < 30 and total>=200:
    total = total - (total * 0.1)
print('จำนวนผู้โดยสารทั้งหมด = ',all)
print('ราคาค่าโดยสาร = ',total)
```

4. การตรวจสอบและประเมินผล

ทดลองรันโปรแกรม แล้วตรวจสอบว่าผลลัพธ์ตรงกับข้อมูลชุดทดสอบที่เตรียมไว้ในขั้นการวิเคราะห์หรือไม่ ถ้าไม่ถูกต้อง ให้ย้อนกลับไปตรวจสอบความถูกต้องของแต่ละขั้นตอนตั้งแต่แรก

ใบกิจกรรมที่ 8.2

การประยุกต์ใช้งาน

กิจกรรมท้ายบท

1.การวิเคราะห์และกำหนดรายละเอียดของปัญหา

ข้อมูลเข้า คือจำนวนจาน.....

ข้อมูลออก คือ จำนวนกล่อง.....

วิธีการตรวจสอบความถูกต้องมีดังนี้

จำนวนจาน : 110

จะต้องเตรียมกล่องทั้งหมด 4 ใบ

ข้อมูลทดสอบที่	ข้อมูลเข้า	ผลลัพธ์
1	จำนวนจาน 305	11
2	จำนวนจาน 130	5
3	จำนวนจาน 5	1
4	จำนวนจาน 110	4

2.การวางแผนการแก้ปัญหา

กล่องแต่ละใบ บรรจุจานได้ 30 ใบ ดังนั้น ถ้านำ 30 ไปหารจำนวนจานที่ล้าง โดยปัดเศษขึ้นให้เป็นจำนวนเต็ม จะได้ผลลัพธ์เป็นจำนวนกล่องที่ต้องใช้

3. การดำเนินการแก้ปัญหา

```

plates=int(input("จำนวนจาน "))
box=0
bit=plates % 30
if bit>0:
    box=(plates-bit)//30 + 1
else:
    box=(plates-bit)//30
print("จำนวนกล่อง",box)

```

4. การตรวจสอบและประเมินผล

ทดลองรันโปรแกรม แล้วตรวจสอบว่าผลลัพธ์ตรงกับข้อมูลชุดทดสอบที่เตรียมไว้ในขั้นการวิเคราะห์หรือไม่ ถ้าไม่ถูกต้อง ให้ย้อนกลับไปตรวจสอบความถูกต้องของแต่ละขั้นตอนตั้งแต่แรก

แบบฝึกหัดท้ายบทข้อที่1.....

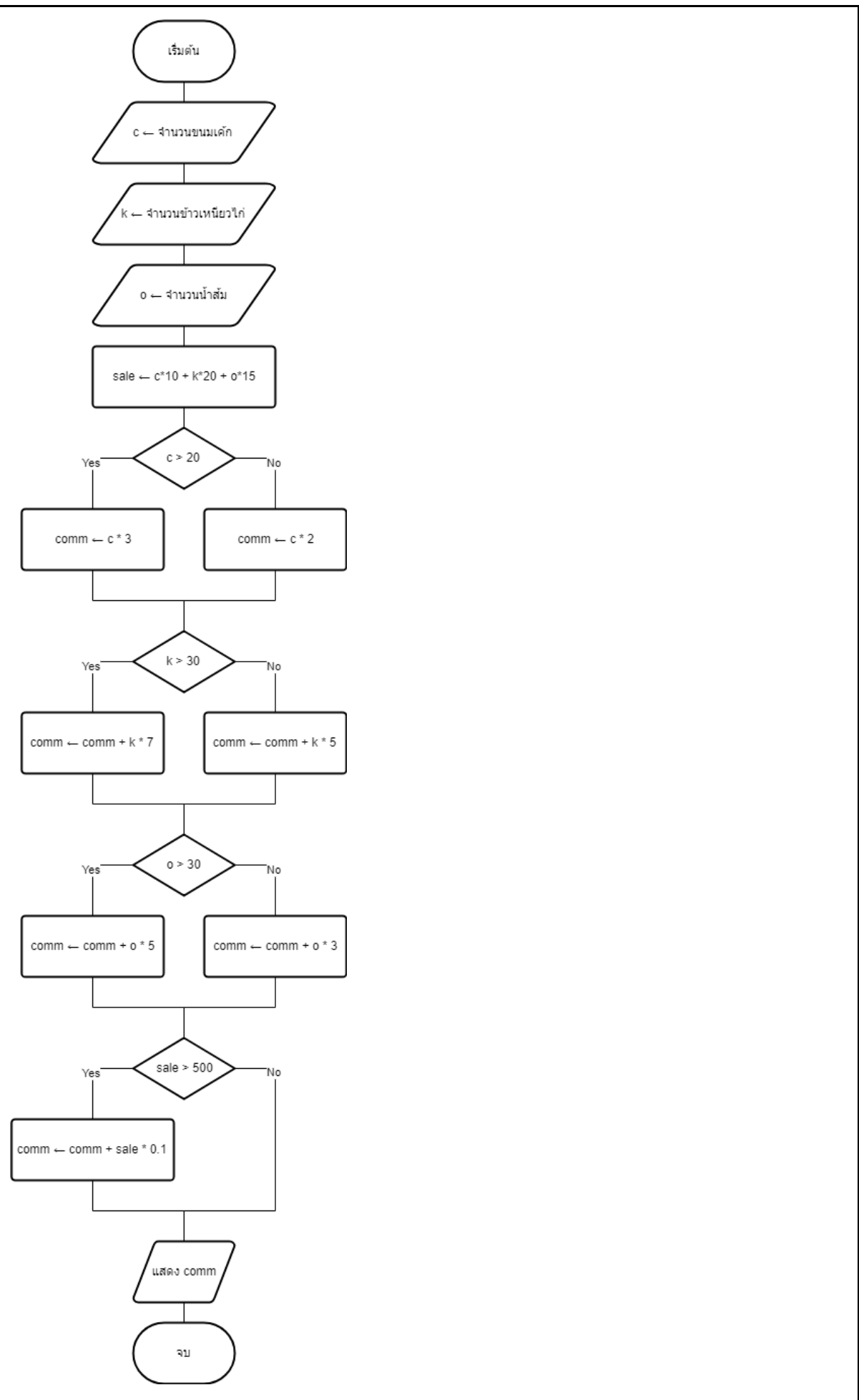
1.การวิเคราะห์และกำหนดรายละเอียดของปัญหา

- 1) ข้อมูลเข้า คือจำนวนขนมเค้ก จำนวนข้าวเหนียวไก่ จำนวนน้ำส้มที่ขายได้.....
- 2) ข้อมูลออก คือ รายได้รวมและส่วนแบ่งที่ได้รับ.....
- 3) วิธีการตรวจสอบความถูกต้อง

ข้อมูลทดสอบที่	ข้อมูลเข้า	ผลลัพธ์
1	จำนวนขนมเค้ก : <u>10</u> จำนวนข้าวเหนียวไก่ : <u>40</u> จำนวนน้ำส้ม : <u>20</u>	รายได้รวม : 1200 ส่วนแบ่งที่ได้รับ : 480
2	จำนวนขนมเค้ก : <u>30</u> จำนวนข้าวเหนียวไก่ : <u>40</u> จำนวนน้ำส้ม : <u>40</u>	รายได้รวม : 1700 ส่วนแบ่งที่ได้รับ : 740
3	จำนวนขนมเค้ก : <u>5</u> จำนวนข้าวเหนียวไก่ : <u>9</u> จำนวนน้ำส้ม : <u>10</u>	รายได้รวม : 380 ส่วนแบ่งที่ได้รับ : 85

4	จำนวนนมเด็ก : <u>12</u> จำนวนข้าวเหนียวไก่ : <u>18</u> จำนวนน้ำส้ม : <u>22</u>	ส่วนแบ่งที่ได้รับ : 261
5	จำนวนนมเด็ก : <u>5</u> จำนวนข้าวเหนียวไก่ : <u>3</u> จำนวนน้ำส้ม : <u>2</u>	ส่วนแบ่งที่ได้รับ : 31
6	จำนวนนมเด็ก : <u>29</u> จำนวนข้าวเหนียวไก่ : <u>4</u> จำนวนน้ำส้ม : <u>4</u>	ส่วนแบ่งที่ได้รับ : 119
7	จำนวนนมเด็ก : <u>33</u> จำนวนข้าวเหนียวไก่ : <u>40</u> จำนวนน้ำส้ม : <u>65</u>	ส่วนแบ่งที่ได้รับ : 914.5

2.การวางแผนการแก้ปัญหา



3. การดำเนินการแก้ปัญหา

```
c=(input (จำนวนขนมเค้กที่ขายได้ '))
k=(input ('จำนวนข้าวเหนียวไก่ที่ขายได้ '))
o=int(input ('จำนวนน้ำส้มที่ขายได้ '))
sale = c*10 + k*20 + o*15
if c>20:
    comm = c*3
else:
    comm = c*2
if k>30:
    comm = comm + k*7
else:
    comm = comm + k*5
if o>30:
    comm = comm + o*5
else:
    comm = comm + o*3
if sale>500:
    comm = comm + sale*0.1
print('ส่วนแบ่งที่ได้ ', comm, 'บาท')
```

4. การตรวจสอบและประเมินผล

.....

ทดลองรันโปรแกรม แล้วตรวจสอบว่าผลลัพธ์ตรงกับข้อมูลชุดทดสอบที่เตรียมไว้ในขั้นการวิเคราะห์หรือไม่ ถ้าไม่ถูกต้อง ให้ย้อนกลับไปตรวจสอบความถูกต้องของแต่ละขั้นตอนตั้งแต่แรก

.....

.....

.....

.....

.....

แบบฝึกหัดท้ายบทข้อที่**2**.....

1.การวิเคราะห์และกำหนดรายละเอียดของปัญหา

- 1) ข้อมูลเข้า คือจำนวนผู้ร่วมเดินทาง.....ระยะทางที่ต้องเดินทางหากไปโดยรถโดยสารประจำทาง และระยะทางหากเดินทางด้วยรถรับจ้างสาธารณะ
- 2) ข้อมูลออก คือ รูปแบบการเดินทาง และราคาค่าโดยสารที่ถูกที่สุด.....
- 3) วิธีการตรวจสอบความถูกต้องมีดังนี้
- การตรวจสอบความถูกต้องของวิธีแก้ปัญหาที่จะออกแบบ คือ ทดลองคิดตัวอย่างการเดินทางที่เป็นไปได้หลาย ๆ รูปแบบ ที่ระยะทางต่าง ๆ กัน แล้วคำนวณหาคำตอบ เอาไว้สำหรับทดสอบวิธีแก้ปัญหาที่ได้ โดยอาจกำหนดในรูปแบบตารางดังนี้

ชุดตรวจสอบที่	จำนวนผู้โดยสาร	ระยะทางโดยรถโดยสาร	ระยะทางโดยรถรับจ้าง	ค่าโดยสารโดยรถโดยสาร	ค่าโดยสารโดยรถรับจ้าง	วิธีที่เลือก
1	1	30	28	150	85	รถรับจ้าง
2	1	5	5	25	39	รถโดยสาร

3	4	5	5	100	39	รถรับจ้าง
4	5	15	12	375	106	รถรับจ้าง
5	10		10		147	รถรับจ้าง
6	1	3			15	รถโดยสาร
7	4		4		37	รถรับจ้าง
8	2		15		59	รถรับจ้าง

2. การวางแผนการแก้ปัญหา

รหัสจำลอง

$n \leftarrow$ จำนวนคนที่ต้องเดินทาง

$d_{\text{Bus}} \leftarrow$ ระยะทางที่เดินทางด้วยรถโดยสาร

$d_{\text{Taxi}} \leftarrow$ ระยะทางที่เดินทางด้วยรถรับจ้างสาธารณะ

คำนวณค่าโดยสารเมื่อเดินทางด้วยรถโดยสาร

$f_{\text{Bus}} \leftarrow n * \text{ceil}(d_{\text{Bus}}) * 5$

คำนวณค่าโดยสารเมื่อเดินทางด้วยรถรับจ้างสาธารณะ

$\text{numTaxi} \leftarrow \text{ceil}(n/4)$

$f_{\text{Taxi}} \leftarrow (35 + \text{ceil}(d_{\text{Taxi}}-3) * 2) * \text{numTaxi}$

ถ้า $f_{\text{Taxi}} \leq f_{\text{Bus}}$

แสดงข้อความ "เดินทางด้วยรถรับจ้างสาธารณะ ค่าโดยสาร", f_{Taxi}

ไม่เช่นนั้น

แสดงข้อความ "เดินทางด้วยรถโดยสารประจำทาง ค่าโดยสาร", f_{Bus}

3. การดำเนินการแก้ปัญหา

สามารถเขียนเป็นโปรแกรมไพทอนได้ดังนี้

```
import math

def busFare(d,n):
    return n*math.ceil(d)*5

def taxiFare(d,n):
    numTaxi=math.ceil(n/4)
    return (35+math.ceil(d-3)*2)*numTaxi

n=int(input('จำนวนคนที่ต้องเดินทาง '))
dBus=int(input('ระยะทางที่เดินทางด้วยรถโดยสาร '))
dTaxi=int(input('ระยะทางที่เดินทางด้วยรถรับจ้างสาธารณะ '))
fBus=busFare(dBus,n)
fTaxi=taxiFare(dTaxi,n)
print('fBus=%.2f, fTaxi=%.2f' % (fBus,fTaxi))
if fTaxi <= fBus:
    print('เดินทางด้วยรถรับจ้างสาธารณะ ค่าโดยสาร', fTaxi, ' บาท')
else:
    print('เดินทางด้วยรถโดยสารประจำทาง ค่าโดยสาร', fBus, ' บาท')
```

4. การตรวจสอบและประเมินผล

ทดลองรันโปรแกรม แล้วตรวจสอบว่าผลลัพธ์ตรงกับข้อมูลชุดทดสอบที่เตรียมไว้ในขั้นการวิเคราะห์หรือไม่ ถ้าไม่ถูกต้อง ให้ย้อนกลับไปตรวจสอบความถูกต้องของแต่ละขั้นตอนตั้งแต่แรก

แบบฝึกหัดท้ายบทข้อที่3.....

1. การวิเคราะห์และกำหนดรายละเอียดของปัญหา

1) ข้อมูลเข้า คือ ส่วนสูง น้ำหนัก และเพศ ของนักเรียนแต่ละคน

2) ข้อมูลออก คือ ค่าเฉลี่ย ส่วนสูง น้ำหนัก ของทั้งห้อง และแยกตามเพศ

3) วิธีการตรวจสอบความถูกต้องคือ ทดลองคิดตัวอย่างของชุดข้อมูลดังกล่าว ไว้สำหรับทดสอบวิธีแก้ปัญหาที่ได้
เช่น

ชุด ตรวจสอบที่	ส่วนสูง	น้ำหนัก	เพศ	ค่าเฉลี่ยส่วน สูงทั้งห้อง	ค่าเฉลี่ยน้ำ หนักทั้งห้อง	ค่าเฉลี่ย ส่วนสูง แยกตาม เพศ	ค่าเฉลี่ยน้ำ หนักแยก ตามเพศ
1							
2							
3							
4							

2. การวางแผนการแก้ปัญหา

หลักการทำงาน คือ ให้รับส่วนสูงของผู้เรียน แล้วตรวจสอบว่าเป็นส่วนสูงที่ถูกต้องหรือไม่ ถ้าไม่ถูกต้อง
ถือว่าผู้ใช้ต้องการออกจากโปรแกรม แต่ถ้าถูกต้องให้ทำงานต่อไป ด้วยการรับน้ำหนัก และเพศ พร้อมกับ เก็บผล
รวมของส่วนสูงและน้ำหนัก แยกตามเพศ เมื่อหมดข้อมูลจึงคำนวณและพิมพ์ผลลัพธ์

รหัสจำลอง

$h \leftarrow$ ส่วนสูง

วนซ้ำถ้า $h > 0$

$w \leftarrow$ น้ำหนัก

$s \leftarrow$ เพศ

ถ้า $s == 'm'$ หรือ $'M'$

```

sumMaleH ← sumMaleH + h
sumMaleW ← sumMaleW + w
numMale ← numMale + 1
ไม่เช่นนั้น
sumFemaleH ← sumFemaleH + h
sumFemaleW ← sumFemaleW + w
numFemale ← numFemale + 1
h ← ส่วนสูง
ถ้า numMale>0
    แสดงค่า 'ส่วนสูง และน้ำหนักเฉลี่ยเพศชาย', sumMaleH/numMale, sumMaleW/numMale
ถ้า numFemale>0
    แสดงค่า 'ส่วนสูง และน้ำหนักเฉลี่ยเพศหญิง', sumFemaleH/numFemale,
sumFemaleW/numFemale
ถ้า numMale+numFemale>0
    แสดงค่า 'ส่วนสูง และน้ำหนักเฉลี่ยทั้งห้อง',
(sumMaleH+sumFemaleH)/(numMale+numFemale),
(sumMaleW+sumFemaleW)/(numMale+numFemale)

```

3. การดำเนินการแก้ปัญหา

สามารถเขียนเป็นโปรแกรมไพทอนได้ดังนี้

```

sumMaleH=0
sumFemaleH=0
sumMaleW=0
sumFemaleW=0
numMale=0
numFemale=0
h=int(input('ส่วนสูง (ป้อน 0 ถ้าหมดข้อมูล) '))
while h>0:
    w=int(input('น้ำหนัก '))
    s=input('เพศ (ป้อน m ผู้ชาย, f ผู้หญิง) ')
    if s=='m':

```

```

        sumMaleH = sumMaleH + h
        sumMaleW = sumMaleW + w
        numMale = numMale + 1
    else:
        sumFemaleH = sumFemaleH + h
        sumFemaleW = sumFemaleW + w
        numFemale = numFemale + 1
    h=int(input('ส่วนสูง (ป้อน 0 ถ้าหมดข้อมูล) '))
if numMale>0:
    print('ส่วนสูง และน้ำหนักเฉลี่ยเพศชาย', sumMaleH/numMale,
sumMaleW/numMale)
if numFemale>0:
    print('ส่วนสูง และน้ำหนักเฉลี่ยเพศหญิง', sumFemaleH/numFemale,
sumFemaleW/numFemale)
if numMale+numFemale>0:
    print('ส่วนสูง และน้ำหนักเฉลี่ยทั้งห้อง',
(sumMaleH+sumFemaleH) / (numMale+numFemale),
(sumMaleW+sumFemaleW) / (numMale+numFemale))

```

4. การตรวจสอบและประเมินผล

ทดลองรันโปรแกรม แล้วตรวจสอบว่าผลลัพธ์ตรงกับข้อมูลชุดทดสอบที่เตรียมไว้ในขั้นการวิเคราะห์หรือไม่ ถ้าไม่ถูกต้อง ให้ย้อนกลับไปตรวจสอบความถูกต้องของแต่ละขั้นตอนตั้งแต่แรก

แบบฝึกหัดข้อ 4

1. การวิเคราะห์และกำหนดรายละเอียดของปัญหา

ข้อมูลเข้า คือ จำนวนกล่อง

ข้อมูลออก คือ ราคาวัตถุดิบที่ใช้ในการทำ และค่าจ้างในการทำ

วิธีการตรวจสอบความถูกต้อง คือ ทดลองคิดตัวอย่างของชุดข้อมูลดังกล่าว ไว้สำหรับทดสอบวิธีแก้ปัญหาก็ได้ เช่น

ข้าวกระเพาะจำนวน 15 กล่อง ราคาวัตถุดิบ 225.5 บาท และค่าจ้างในการทำ 67.65 บาท

ข้าวกระเพาะจำนวน 30 กล่อง ราคาวัตถุดิบ 451 บาท และค่าจ้างในการทำ 134.5 บาท

2. การวางแผนการแก้ปัญหา

นำราคาวัตถุดิบมาคิดในอัตราส่วนที่ใช้จริงแล้วนำมาบวกกันจนครบ เช่น

ราคาวัตถุดิบ	ปริมาณที่ใช้ต่อข้าวกระเพรา 15 กล่อง	ราคาต่อปริมาณที่ใช้
ข้าวหอมมะลิ ราคา 38 บาท/กิโลกรัม	2 กิโลกรัม	76
เนื้อไก่ ราคา 70 บาท/โลกรัม	1 กิโลกรัม	70
ใบกระเพรา ราคา 20 บาท/กิโลกรัม	200 กรัม	2
พริก ราคา 45 บาท/กิโลกรัม	200 กรัม	9
กระเทียม ราคา 80 บาท/กิโลกรัม	200 กรัม	16
ไข่ไก่ 3.5 บาท/ฟอง	15 ฟอง	52.5
	รวม	225.5

หลังจากนั้นนำราคาวัตถุดิบที่ใช้ทั้งหมดหารได้จำนวนกล่องที่สามารถทำข้าวกระเพราไก่ได้ ในที่นี้คือ 15 กล่อง

รหัสจำลอง

เริ่มต้น

1. `pricePerBox <- 225.5/15`
2. `number ← รับจำนวนข้าวกระเพราที่ต้องการ (กล่อง)`
3. `ค่าวัตถุดิบที่ต้องใช้ ← pricePerBox*number`
4. `ค่าจ้างของป้าเชอรี่ ← ค่าวัตถุดิบที่ต้องใช้ * 0.3`

จบ

3. การดำเนินการแก้ปัญหา เขียนโปรแกรมไพทอนได้ดังนี้

```
foodPricePerBox = 225.5/15
```

```
number = 0
```

```
number=int(input('จำนวนข้าวกระเพรากี่กล่อง '))
```

```
total=number*foodPricePerBox
```

```
print('ค่าวัตถุดิบที่ต้องใช้ =%5d บาท ,ค่าจ้างของป้าเชอรี่ =%5.2f บาท ' %  
(total,total*0.3))
```