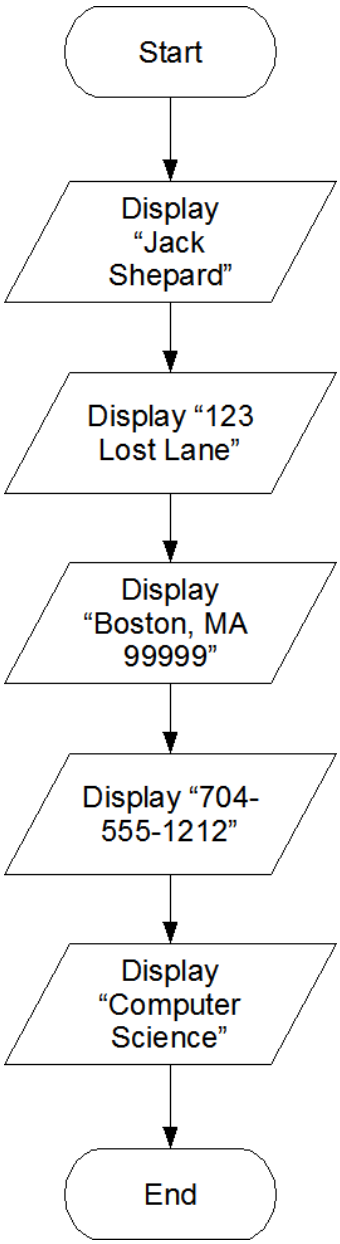
Starting Out With Python 2nd Edition Gaddis Solutions Manual
Full Download: http://alibabadownload.com/product/starting-out-with-python-2nd-edition-gaddis-solutions-manual/

Programming Exercise 2-1

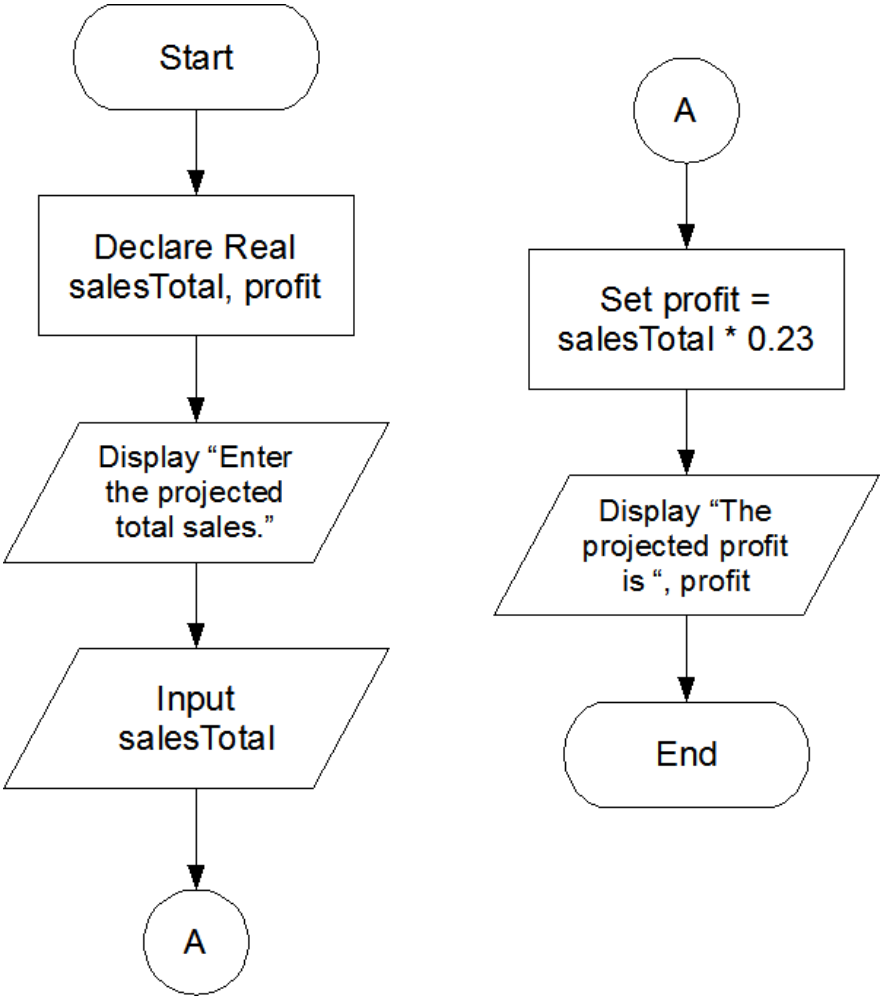
```
print ("Jack Shepard")
print ("123 Lost Lane")
print ("Boston, MA  99999")
print ("704-555-1212")
print ("Computer Science")
```



This sample only, Download all chapters at: alibabadownload.com

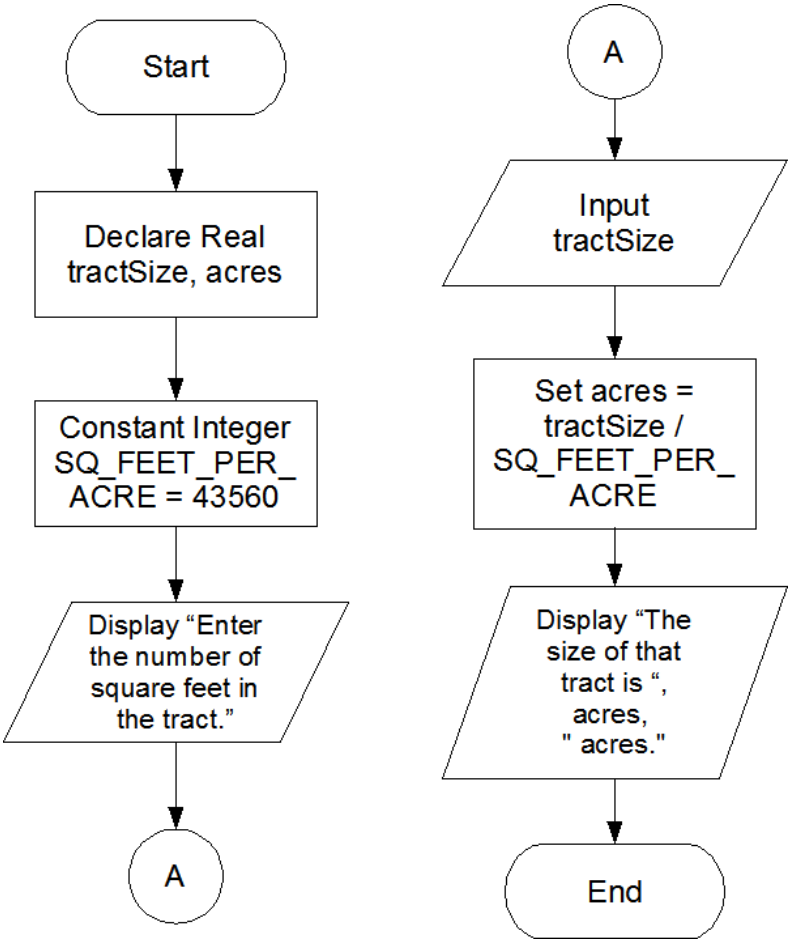
Programming Exercise 2-2

```
# Variables to hold the sales total and the profit
salesTotal = 0.0
profit = 0.0

# Get the amount of projected sales.
salesTotal = float(input("Enter the projected sales: "))

# Calculate the projected profit.
profit = salesTotal * 0.23

# Print the projected profit.
print ("The projected profit is ", format(profit, '.2f'))
```



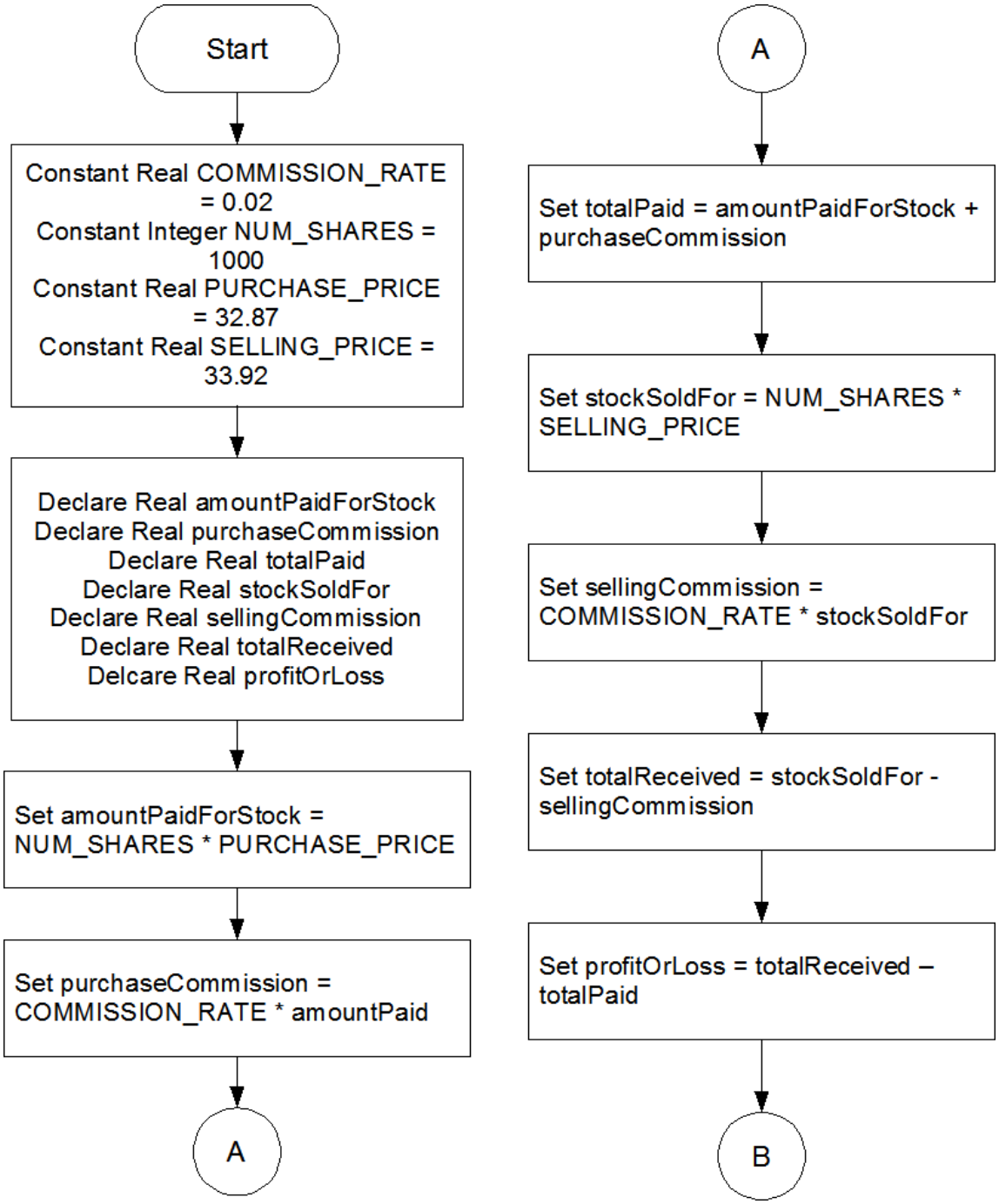
Programming Exercise 2-3

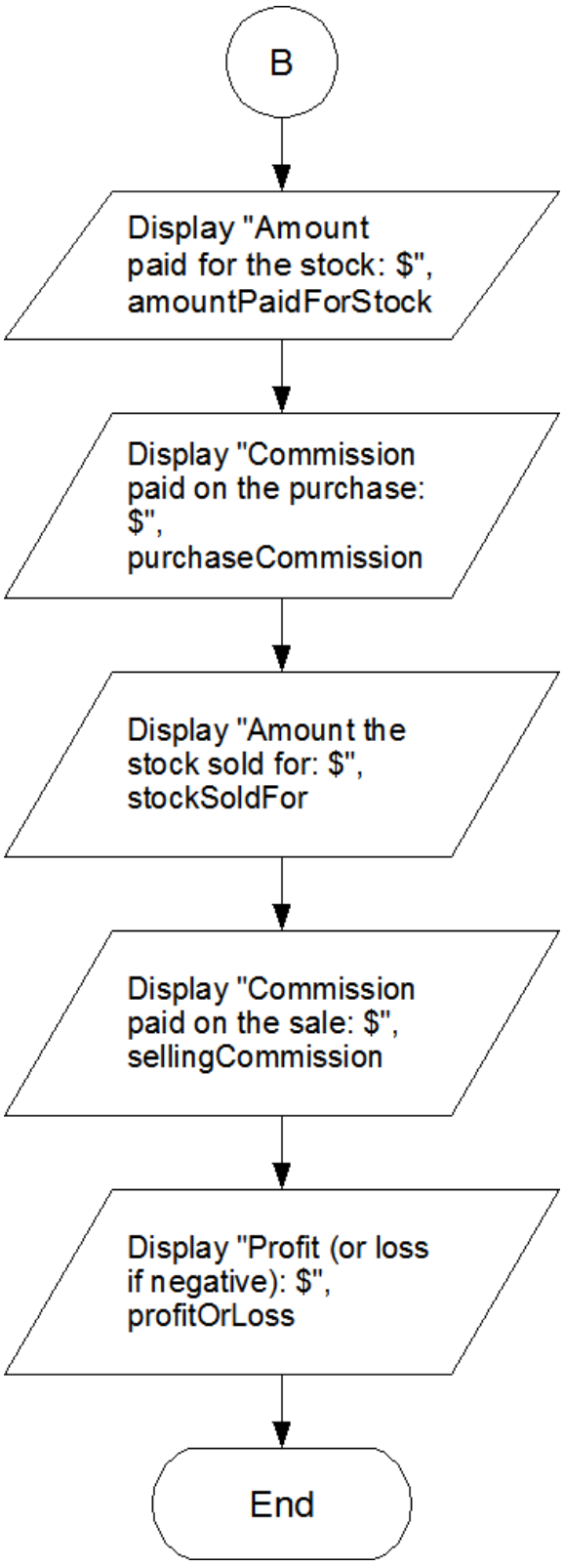
```
# Variables to hold the size of the tract and number of acres.
tractSize = 0.0
acres = 0.0

# Constant for the number of square feet in an acre.
SQ_FEET_PER_ACRE = 43560

# Get the square feet in the tract.
tractSize = input("Enter the number of square feet in the tract.")

# Calculate the acreage.
acres = float(tractSize) / SQ_FEET_PER_ACRE

# Print the number of acres.
print ("The size of that tract is", format(acres, '.2f'), "acres.")
```



Programming Exercise 2-10

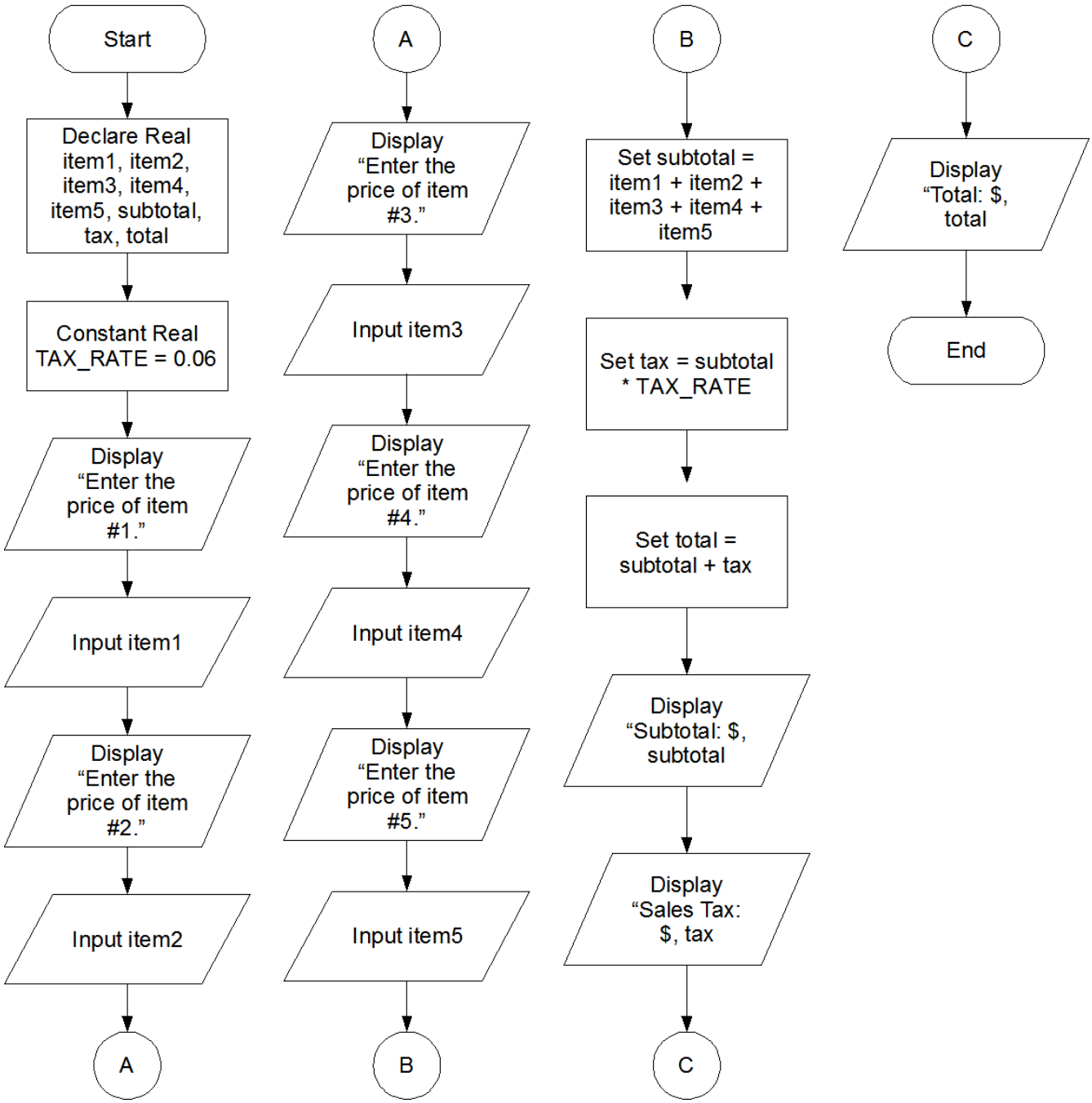
```
# Named constants
COMMISSION_RATE = 0.02
NUM_SHARES = 1000
PURCHASE_PRICE = 32.87
SELLING_PRICE = 33.92

# Variables
amountPaidForStock = 0.0  # Amount paid for the stock
purchaseCommission = 0.0  # Commission paid to purchase stock
totalPaid = 0.0           # Total amount paid
stockSoldFor = 0.0        # Amount stock sold for
sellingCommission = 0.0   # Commission paid to sell stock
totalReceived = 0.0       # Total amount received
profitOrLoss = 0.0        # Amount of profit or loss

# Calculate the amount that Joe paid for the stock, not
# including the commission.
amountPaidForStock = NUM_SHARES * PURCHASE_PRICE

# Calculate the amount of commission that Joe paid his broker
# when he bought the stock.
purchaseCommission = COMMISSION_RATE * amountPaidForStock

# Calculate the total amount that Joe paid, which is the amount
# he paid for the stock plus the commission he paid his broker.
totalPaid = amountPaidForStock + purchaseCommission

# Calcualate the amount that Joe sold the stock for.
stockSoldFor = NUM_SHARES * SELLING_PRICE

# Calculate the amount of commission that Joe paid his broker
# when he sold the stock.
sellingCommission = COMMISSION_RATE * stockSoldFor

# Calculate the amount of money left over, after Joe paid
# his broker.
totalReceived = stockSoldFor - sellingCommission

# Calculate the amount of profit or loss. If this amount is a
# positive number, it is profit. If this is a negative number it
# is a loss.
profitOrLoss = totalReceived – totalPaid

# Print the required data.
print ("Amount paid for the stock: ", format(amountPaidForStock, '.2f'))
print ("Commission paid on the purchase: ", format(purchaseCommission, '.2f'))
print ("Amount the stock sold for: ", format(stockSoldFor, '.2f'))
print ("Commission paid on the sale: ", format(sellingCommission, '.2f'))
print ("Profit (or loss if negative): ", format(profitOrLoss, '.2f'))
```


Start
Constant Real COMMISSION_RATE = 0.02
Constant Integer NUM_SHARES = 1000
Constant Real PURCHASE_PRICE = 32.87
Constant Real SELLING_PRICE = 33.92
Declare Real amountPaidForStock
Declare Real purchaseCommission
Declare Real totalPaid
Declare Real stockSoldFor
Declare Real sellingCommission
Declare Real totalReceived
Delcare Real profitOrLoss
Set amountPaidForStock = NUM_SHARES * PURCHASE_PRICE
Set purchaseCommission = COMMISSION_RATE * amountPaid
A
A
Set totalPaid = amountPaidForStock + purchaseCommission
Set stockSoldFor = NUM_SHARES * SELLING_PRICE
Set sellingCommission = COMMISSION_RATE * stockSoldFor
Set totalReceived = stockSoldFor - sellingCommission
Set profitOrLoss = totalReceived – totalPaid
B


B
Display "Amount paid for the stock: $", amountPaidForStock
Display "Commission paid on the purchase: $", purchaseCommission
Display "Amount the stock sold for: $", stockSoldFor
Display "Commission paid on the sale: $", sellingCommission
Display "Profit (or loss if negative): $", profitOrLoss
End

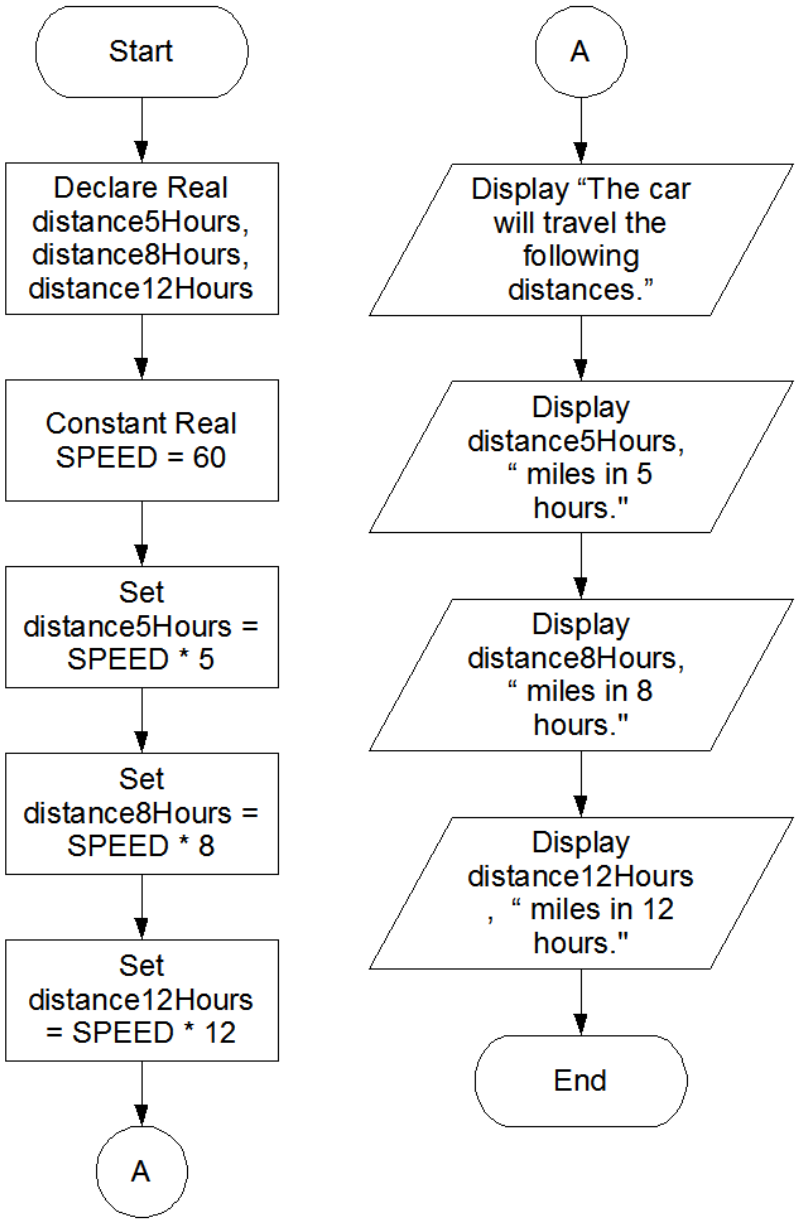
Programming Exercise 2-4

```
# Variables to hold the prices of each item, the subtotal,
# and the total.
item1 = 0.0
item2 = 0.0
item3 = 0.0
item4 = 0.0
item5 = 0.0
subtotal = 0.0
tax = 0.0
total = 0.0

# Constant for the sales tax rate.
TAX_RATE = 0.06

# Get the price of each item.
item1 = float(input("Enter the price of item #1: "))
item2 = float(input("Enter the price of item #2: "))
item3 = float(input("Enter the price of item #3: "))
item4 = float(input("Enter the price of item #4: "))
item5 = float(input("Enter the price of item #5: "))

# Calculate the subtotal.
subtotal = item1 + item2 + item3 + item4 + item5

# Calculate the sales tax.
tax = subtotal * TAX_RATE

# Calculate the total.
total = subtotal + tax

# Print the values.
print ("Subtotal: ", format(subtotal, '.2f'))
print ("Sales Tax: ", format(tax, '.2f'))
print ("Total: ", format(total, '.2f'))
```


Start
Declare Real item1, item2, item3, item4, item5, subtotal, tax, total
Constant Real TAX_RATE = 0.06
Display "Enter the price of item #1."
Input item1
Display "Enter the price of item #2."
Input item2
A
A
Display "Enter the price of item #3."
Input item3
Display "Enter the price of item #4."
Input item4
Display "Enter the price of item #5."
Input item5
B
B
Set subtotal = item1 + item2 + item3 + item4 + item5
Set tax = subtotal * TAX_RATE
Set total = subtotal + tax
Display "Subtotal: $, subtotal
Display "Sales Tax: $, tax
C
C
Display "Total: $, total
End

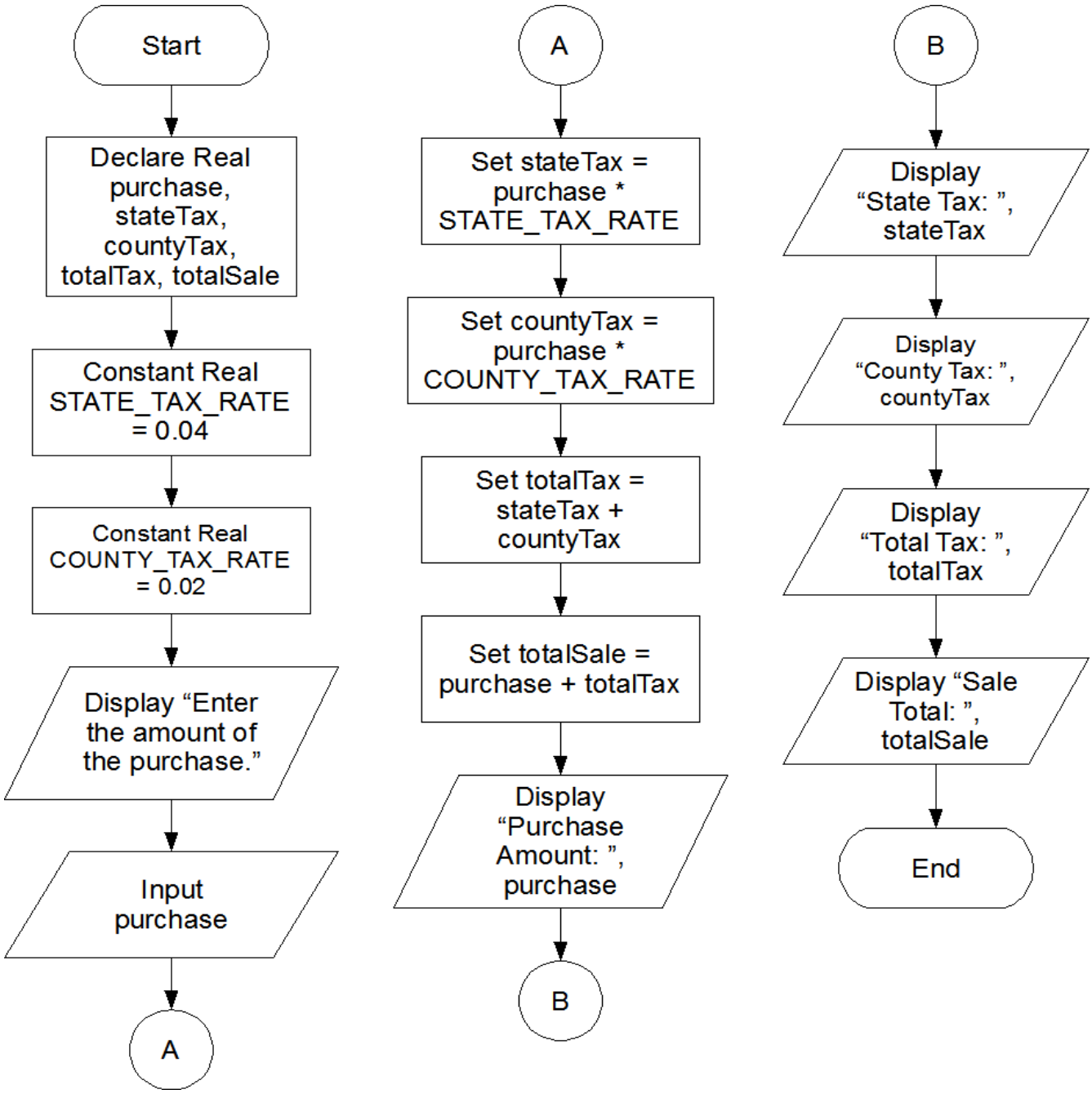
Programming Exercise 2-5

```
# Variables to hold the distances.
distance5Hours = 0.0
distance8Hours = 0.0
distance12Hours = 0.0

# Constant for the speed.
SPEED = 60

# Calculate the distance the car will travel in
# 5, 8, and 12 hours.
distance5Hours = SPEED * 5
distance8Hours = SPEED * 8
distance12Hours = SPEED * 12

# Print the results.
print ("The car will travel the following distances:")
print (distance5Hours," miles in 5 hours.")
print (distance8Hours," miles in 8 hours.")
print (distance12Hours," miles in 12 hours.")
```


Start
Declare Real distance5Hours, distance8Hours, distance12Hours
Constant Real SPEED = 60
Set distance5Hours = SPEED * 5
Set distance8Hours = SPEED * 8
Set distance12Hours = SPEED * 12
A
A
Display “The car will travel the following distances.”
Display distance5Hours, “ miles in 5 hours."
Display distance8Hours, “ miles in 8 hours."
Display distance12Hours , “ miles in 12 hours."
End

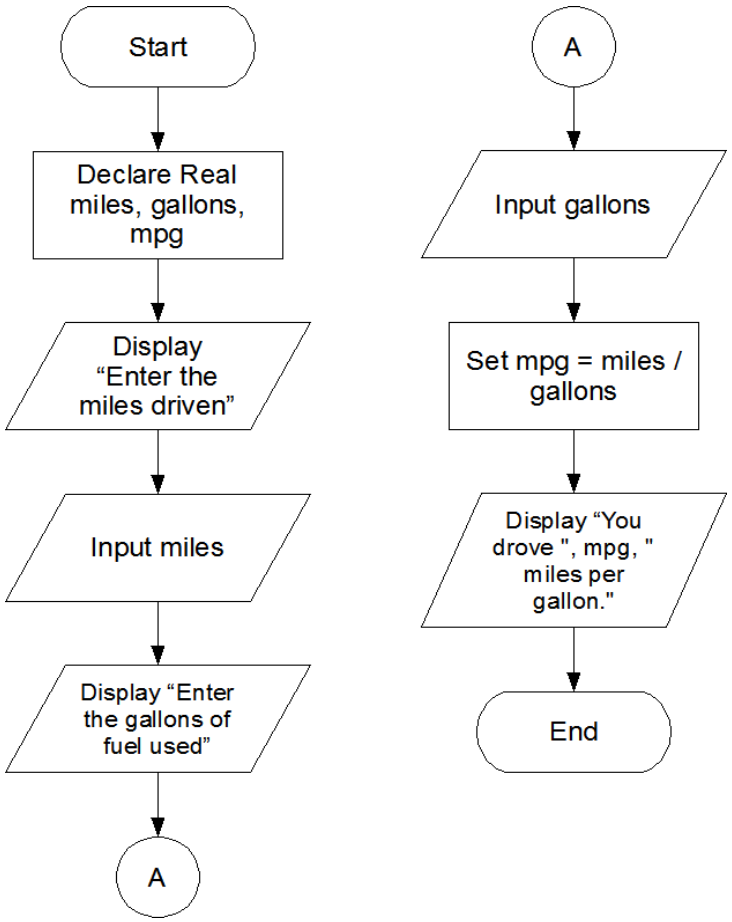
Programming Exercise 2-6

```
# Variable declarations
purchase = 0.0
stateTax = 0.0
countyTax = 0.0
totalTax = 0.0
totalSale = 0.0

# Constants for the state and county tax rates
STATE_TAX_RATE = 0.04
COUNTY_TAX_RATE = 0.02

# Get the amount of the purchase.
purchase = float(input("Enter the amount of the purchase: "))

# Calculate the state sales tax.
stateTax = purchase * STATE_TAX_RATE

# Calculate the county sales tax.
countyTax = purchase * COUNTY_TAX_RATE

# Calculate the total tax.
totalTax = stateTax + countyTax

# Calculate the total of the sale.
totalSale = purchase + totalTax

# Print information about the sale.
print ("Purchase Amount: ", format(purchase, '.2f'))
print ("State Tax: ", format(stateTax, '.2f'))
print ("County Tax: ", format(countyTax, '.2f'))
print ("Total Tax: ", format(totalTax, '.2f'))
print ("Sale total: ", format(totalSale, '.2f'))
```


Start
Declare Real purchase, stateTax, countyTax, totalTax, totalSale
Constant Real STATE_TAX_RATE = 0.04
Constant Real COUNTY_TAX_RATE = 0.02
Display "Enter the amount of the purchase."
Input purchase
A
A
Set stateTax = purchase * STATE_TAX_RATE
Set countyTax = purchase * COUNTY_TAX_RATE
Set totalTax = stateTax + countyTax
Set totalSale = purchase + totalTax
Display "Purchase Amount: ", purchase
B
B
Display "State Tax: ", stateTax
Display "County Tax: ", countyTax
Display "Total Tax: ", totalTax
Display "Sale Total: ", totalSale
End

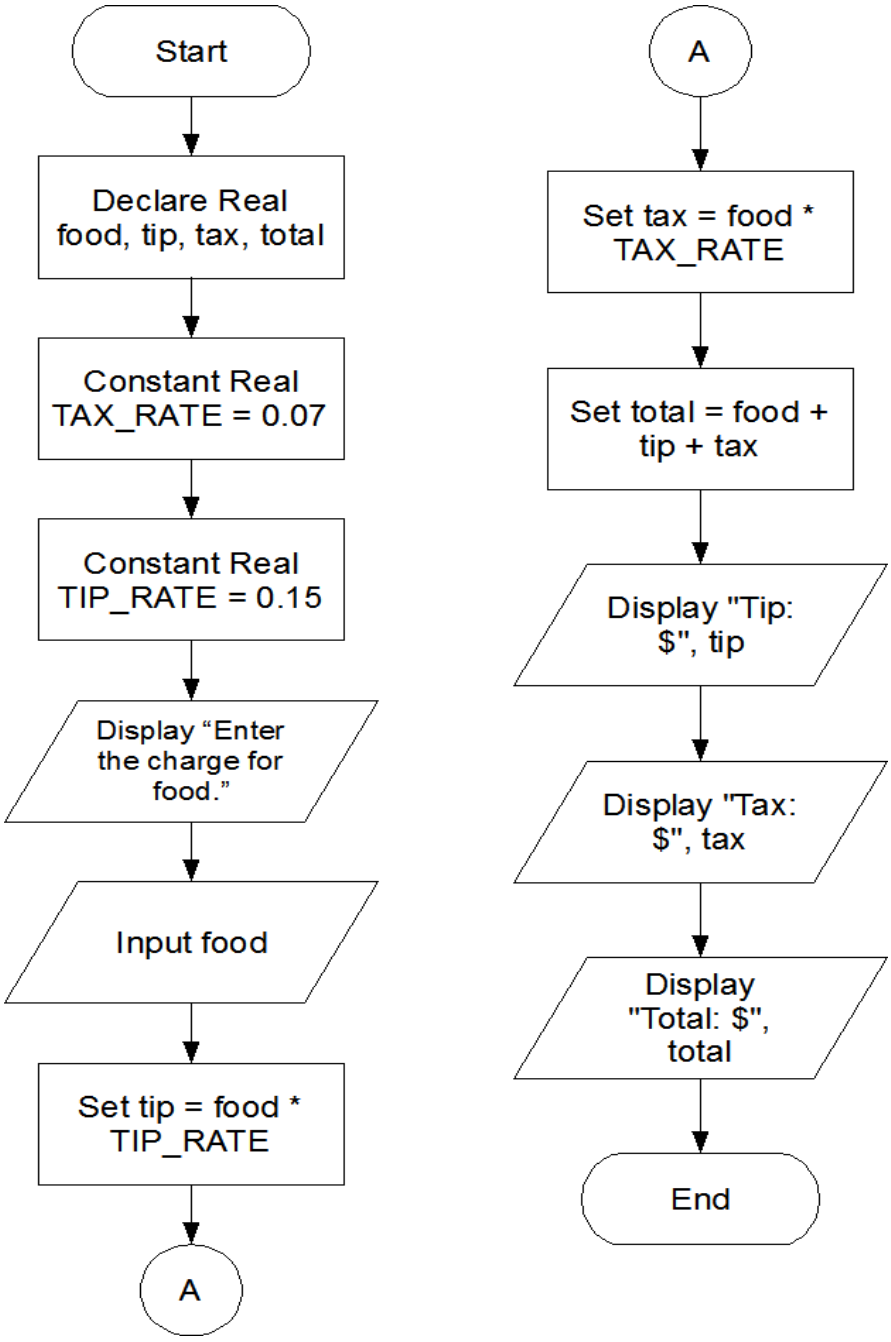
Programming Exercise 2-7

```
# Declare variables to hold miles driven, gallons
# of fuel used, and miles-per-gallon.
miles = 0.0
gallons = 0.0
mpg = 0.0

# Get the miles driven.
miles = float(input("Enter the miles driven: "))

# Get the gallons of fuel used.
gallons = float(input("Enter the gallons of fuel used: "))

# Calculate miles-per-gallon.
mpg = miles / gallons

# Print the result.
print ("You used", format(mpg, '.2f'), "miles per gallon.")
```



Programming Exercise 2-8

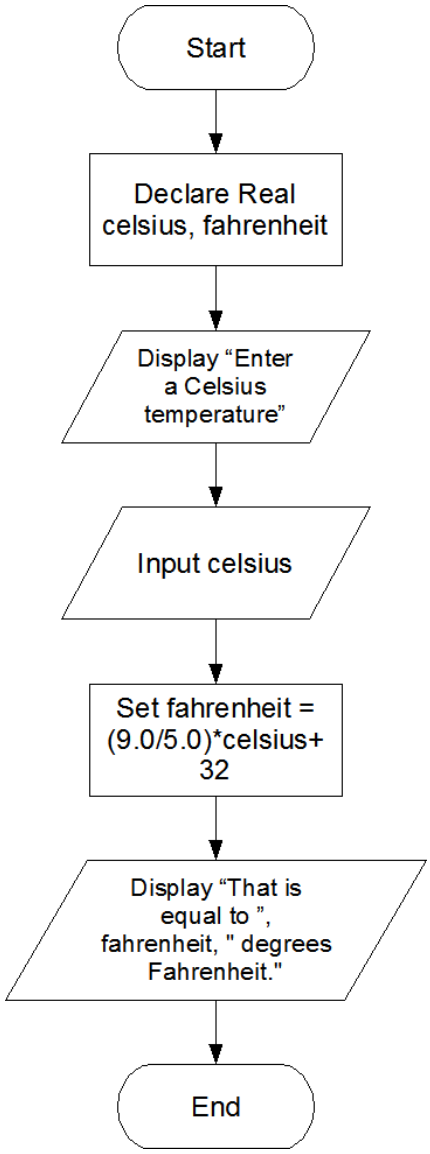
```
# Declare variables for food charges, tip, tax, and total.
food = 0.0
tip = 0.0
tax = 0.0
total = 0.0

# Constants for the tax rate and tip rate.
TAX_RATE = 0.07
TIP_RATE = 0.15

# Get the food charges.
food = float(input("Enter the charge for food: "))

# Calculate the tip.
tip = food * TIP_RATE

# Calculate the tax.
tax = food * TAX_RATE

# Calculate the total.
total = food + tip + tax

# Print the tip, tax, and total.
print ("Tip: ", format(tip, '.2f'))
print ("Tax: ", format(tax, '.2f'))
print ("Total: ", format(total, '.2f'))
```


Start
Declare Real
food, tip, tax, total
Constant Real
TAX_RATE = 0.07
Constant Real
TIP_RATE = 0.15
Display “Enter
the charge for
food.”
Input food
Set tip = food *
TIP_RATE
A
A
Set tax = food *
TAX_RATE
Set total = food +
tip + tax
Display "Tip:
$", tip
Display "Tax:
$", tax
Display
"Total: $",
total
End

Starting Out With Python 2nd Edition Gaddis Solutions Manual
Full Download: http://alibabadownload.com/product/starting-out-with-python-2nd-edition-gaddis-solutions-manual/

Programming Exercise 2-9

```
# Declare variables to hold the temperatures.
celsius = 0.0
fahrenheit = 0.0

# Get the Celsius temperature.
celsius = float(input("Enter a Celsius temperature: "))

# Calculate the Fahrenheit equivalent.
fahrenheit = (9.0 / 5.0) * celsius + 32

# Display the Fahrenheit temperature.
print ("That is equal to", format(fahrenheit, '.2f'), "degrees
Fahrenheit.")
```



This sample only, Download all chapters at: alibabadownload.com