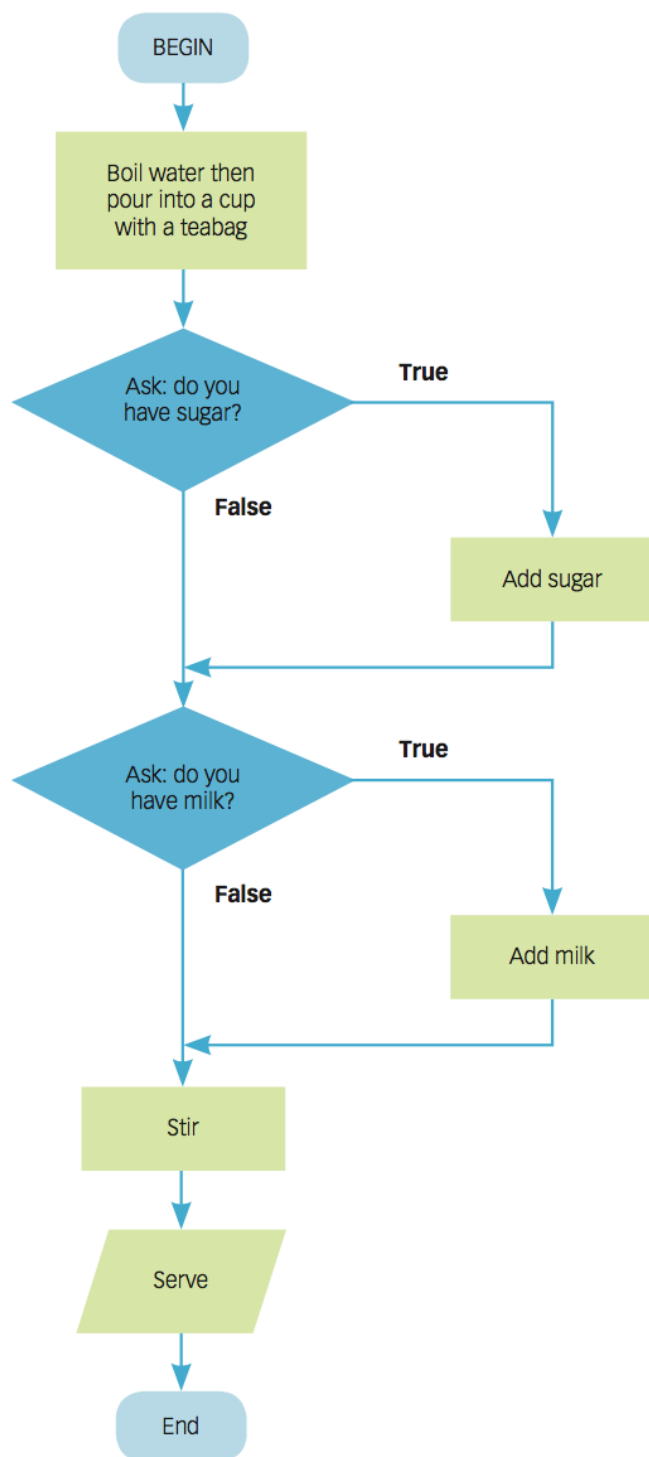


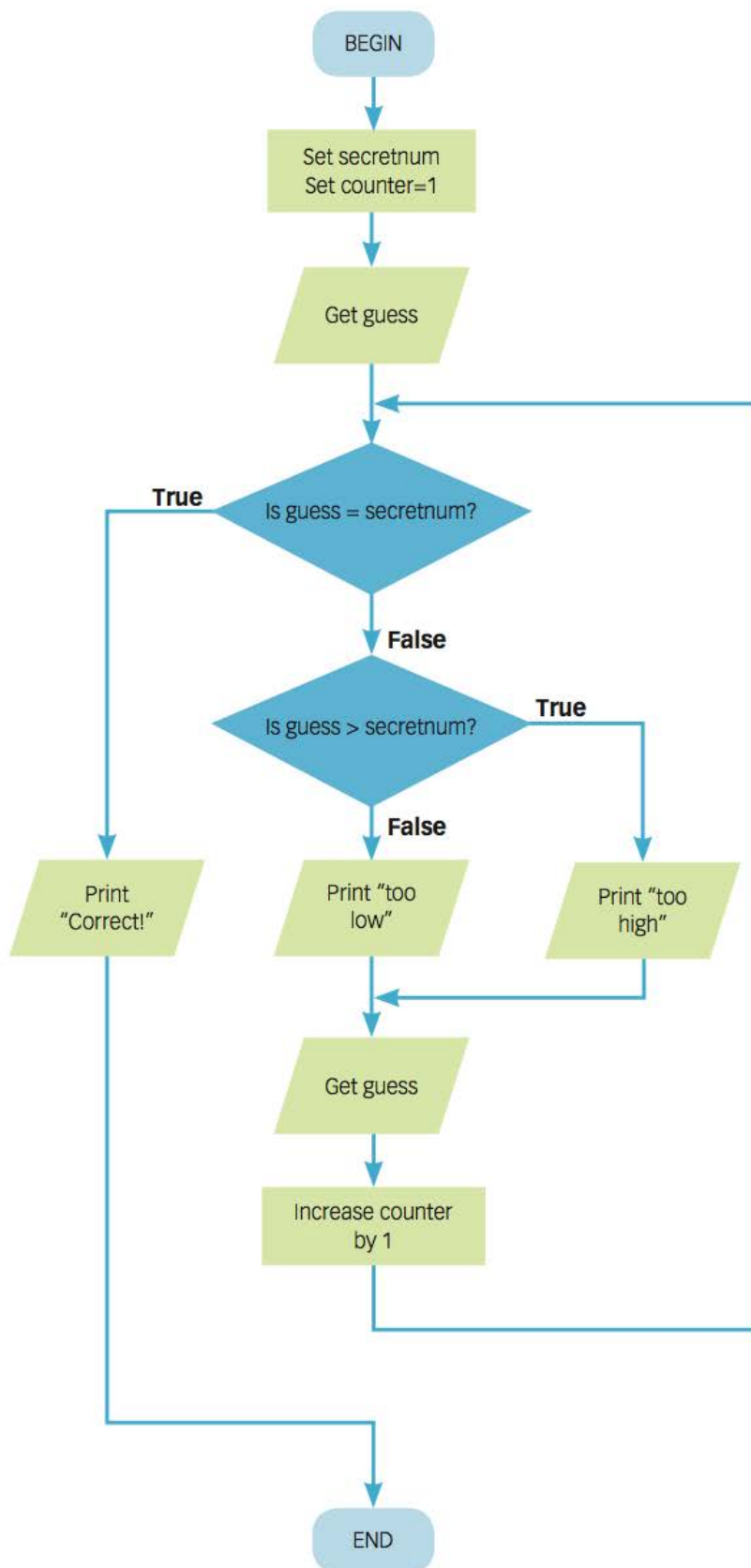
## Solutions

### Chapter 3: Understanding programming using a general-purpose language

Activity: Flowcharts, page 42



## Activity: Number guesser, pages 42–3



## Skill Builder: Version 1, page 49

```
secret = 30
flag = True
while flag == True:
    guess = input('Your guess?: ')
    guess = int(guess)
    if guess > secret:
        print('Too high')
    elif guess < secret:
        print('Too low')
    else:
        print('Correct!')
        flag=False
```

## Skill builder: Version 2, pages 50–1

```
import random
secret = random.randint(1,100)
print(secret)
flag = True
while flag == True:
    guess = input('Your guess?: ')
    guess = int(guess)
    if guess < secret:
        print('Too low')
    elif guess > secret:
        print('Too high')
    else:
        print('Correct!')
        flag=False
```

## Activity: Basics of input and output, page 51

*The following solutions use Python.*

1 print('Hello')

2

```
firstname=input('What is your first name? ')
familyname=input('What is your family name? ')
favsinger=input('Who is your favourite singer? ')
favactor=input('Who is your favourite movie actor? ')
print('Hello! I know a lot about you! You are',firstname,familyname,\
      '- your favourite singer is', favsinger,'and',favactor,'is your favourite movie actor.')
```

3

```
print('Program to calculate the approximate area of a circle')
rad=float(input('What radius in mm is the circle?')) #string must be converted to floating point
area=3.17*rad*rad
print('The approx area is',area,'sq mm.')
```

4

```
a=input("What multiplication table would you like to display? ")
a=int(a)
b=1
while b<=12:
    print(a,"X",b,"=",a*b)
    b=b+1
print("That's all!")
```

5

```
a=1
b=1
while a<=12:
    while b<=12:
        print(a,"X",b,"=",a*b)
        b=b+1
    a=a+1
    b=1
    print()
print("That's all!")
```

## Skill builder: Using programming functions, page 51

*The following solutions use Python.*

1

```
myword=input('Give me a word and I will count the letters. ')
count=len(myword)
print ('The number of letters in',myword,'is',count)
```

- 2 This could be used to correct any errors in titles of chapters of a book or in a large list of people's names following data entry.

```
myword=input('Give me your full name written in lower case. ')
allletters=str.upper(myword)
print(allletters)
firstletters=str.title(myword)
print(firstletters)
```

- 3 There are many methods but here is a simple one. First the sentence is split into separate words, ignoring all spaces. Then these words are rejoined using a single space between each one.

```
sentence=input('Enter the sentence you want me to correct for extra spaces. ')
fixedsentence=" ".join(str.split(sentence))
print(fixedsentence)
```

### Skill builder: Version 3, page 51

The three syntax errors are commented on below in this corrected version.

```
import random
secret = random.randint(1,100)
print(secret)
flag = True
while flag == True:
    guess = int(input('Your guess?: '))    #string must be converted to integer
    if guess < secret:
        print('Too low')
    elif guess > secret:                    #missing colon
        print('Too high')
    else:
        print('Correct!')
        flag=False                        #Keyword False must be capitalised.
```

### Skill builder: Version 4, page 52

The three logic errors are commented on below in this corrected version and the effect on the running of the program is stated for each.

```
import random
secret = random.randint(1,100)
print(secret)
flag = True
while flag == True:                        #Question 3: Flag was set to False. Loop is never entered.
    guess = input('Your guess?: ')
    guess = int(guess)
    if guess > secret:                      #Question 2: High and low statements were reversed. Wrong message delivered.
        print('Too high')
    elif guess < secret:
        print('Too low')
    else:
        print('Correct!')
        flag=False                        #Question 1: Incorrect indentation, so quit after only one guess
```