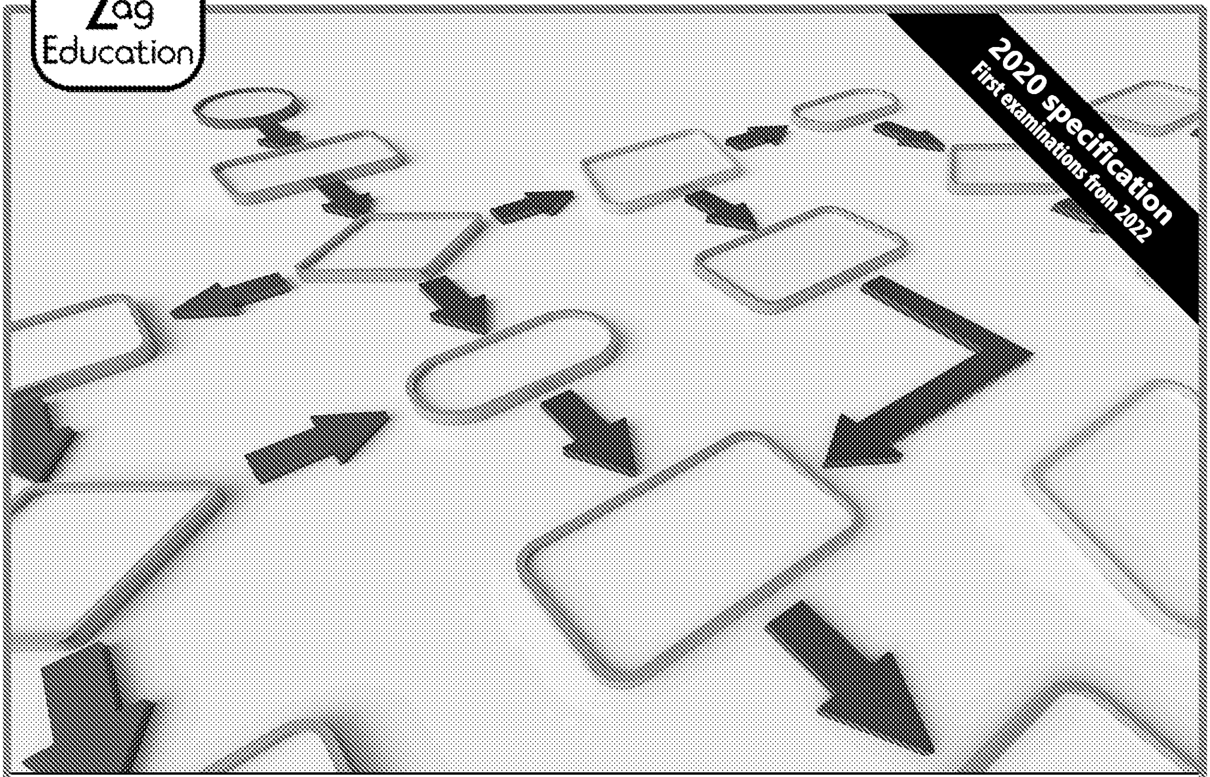
Computer Science GCSE (9–1) | OCR | J277



Algorithms Resource Pack

for OCR GCSE Computer Science (J277)

Sue Wright

Part 2 – Worksheets & Solutions

zigzageducation.co.uk

POD 10603b

Contents

EXERCISE 1: CHARITY FUNDRAISER – ANALYS

Identify the inputs, process and outputs you would need to know to solve this pr

You have been asked to write a simple algorithm to work out how much money
fundraising activity at school and display the total.

The activities your form took part in were:

- Car washing
- Dog walking

For example, you will know how many cars were washed and what the charge w

INPUTS	PROCESS	

EXERCISE 1: CHARITY FUNDRAISER – ANALYS

Identify the inputs, process and outputs you would need to know to solve this pr

You have been asked to write a simple algorithm to work out how much money
fundraising activity at school and display the total.

The activities your form took part in were:

- Car washing
- Dog walking

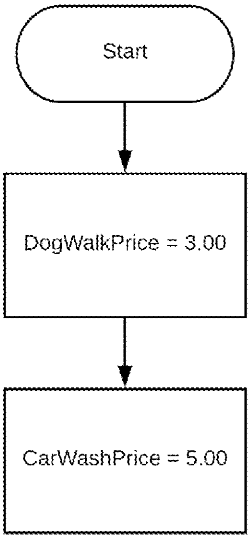
For example, you will know how many cars were washed and what the charge w

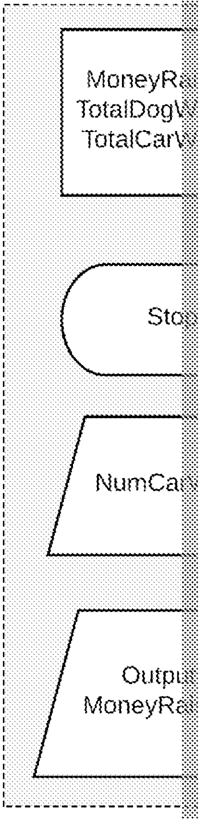
INPUTS	PROCESS	

EXERCISE 2: CHARITY FUNDRAISER – PUT THE CORRECT ORDER

Now that we have identified the inputs, process and outputs needed to solve the chart to give a visual representation of our algorithm. It has been decided that th walking and £5 for car washing.

The flow chart has been started below (on the left); you need to add the remaini correct order.





EXERCISE 3: CONSTANTS OR VARI

Complete the table to identify which of the following are constants and which ar
Fill in the last column to explain your answer.

EXPRESSION	CONSTANT OR VARIABLE?	
currentTemp = 30		
pi = 3.14159		
diameter = 34.5		
boilPoint = 100		
currentShoeSize = 5.5		
daysInWeek = 7		
minsInHour = 60		
playerOneDiceRoll = 5		
gramToOunce = 0.0352		
playerName = "Charlotte"		

EXERCISE 4: HOLIDAY CALCULAT

You have been invited on a four-day holiday to Disneyland Paris with a friend. Th food have been paid for; you need to have money for drinks and souvenirs. You the holiday is a month away so you could have more money by then.

Write an algorithm using **OCR Exam Reference Language** that will calculate how each day. You should start by identifying your inputs, process and outputs befor

INPUTS	PROCESS	

Note: Your answer should show the use of constants, variables, the INPUT() and PRINT() assigning a value to a variable in OCR Exam Reference Language.

EXERCISE 5: HOLIDAY TEMPERATURE C

You are visiting a member of your family, who lives in Florida, for a holiday in De temperature will be about 61 ° Fahrenheit; we use Celsius to measure temperatu

Write an algorithm using OCR Exam Reference Language which will allow the use Fahrenheit and output the equivalent in Celsius to the screen.

*Note: The formula will be (F – 32) * 5/9 = C.*

Identify your inputs, process and outputs first.

INPUTS	PROCESS	

EXERCISE 6: ODDS OR EVENS

Design a simple algorithm that will take in a number from the user and output wh

Hint: a number that is divisible by 2 with no remainder will be even.

Identify your inputs, process and outputs first.

INPUTS	PROCESS	

This should be written **BOTH** in OCR Exam Reference Language and as a flow cha

Flow chart	Pseudo code

EXERCISE 7: COLOUR RANGE

Write an algorithm that will take in a number, check that the number is within a correct colour. If the number is not in the correct range the algorithm must displ

- Between 0 to 10 = red
- Between 11 to 20 = green
- Between 21 to 30 = blue

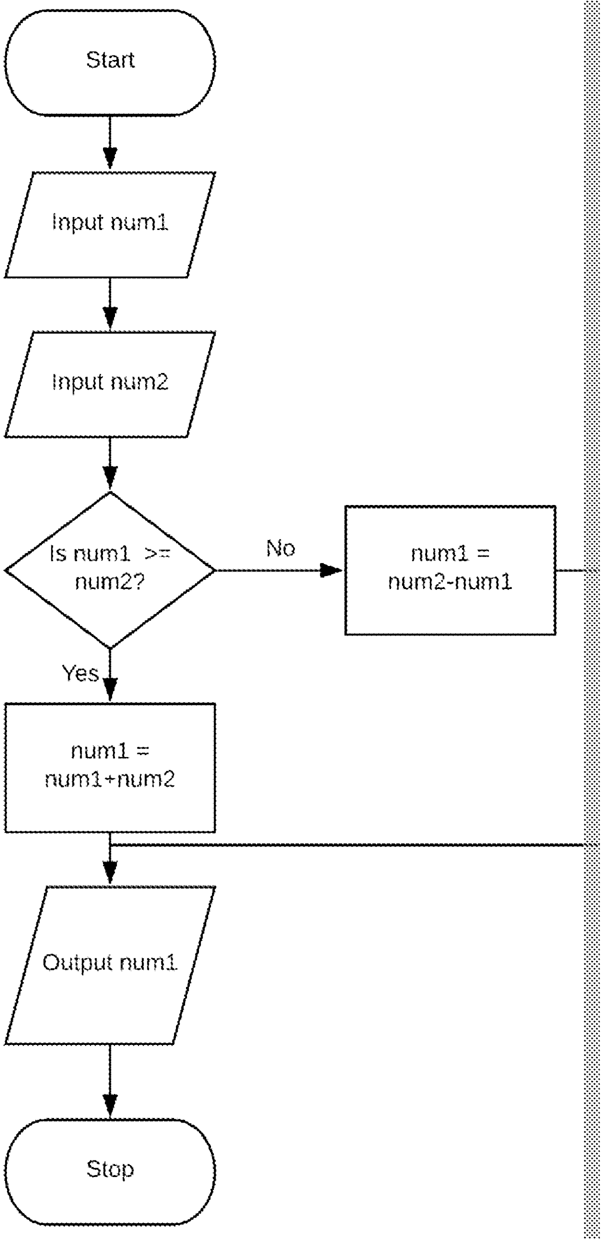
Identify your inputs, process and outputs first, the produce **BOTH** OCR Exam Ref below.

INPUTS	PROCESS	

Flow chart	OCR Exam Refere

EXERCISE 8: TRACE TABLE 1

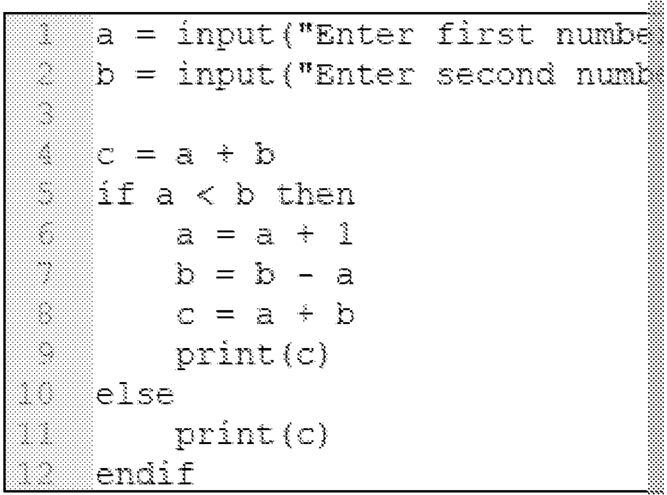
Study the flow chart and complete the trace table below. The first example has b



num1	num2	num1 >= num2	nu
5	9	False	4
3	8		
2	10		
12	5		
1	20		
17	3		

EXERCISE 9: TRACE TABLE 2

Read the OCR Exam Reference Language carefully and complete the trace table b
The first row has been completed for you.

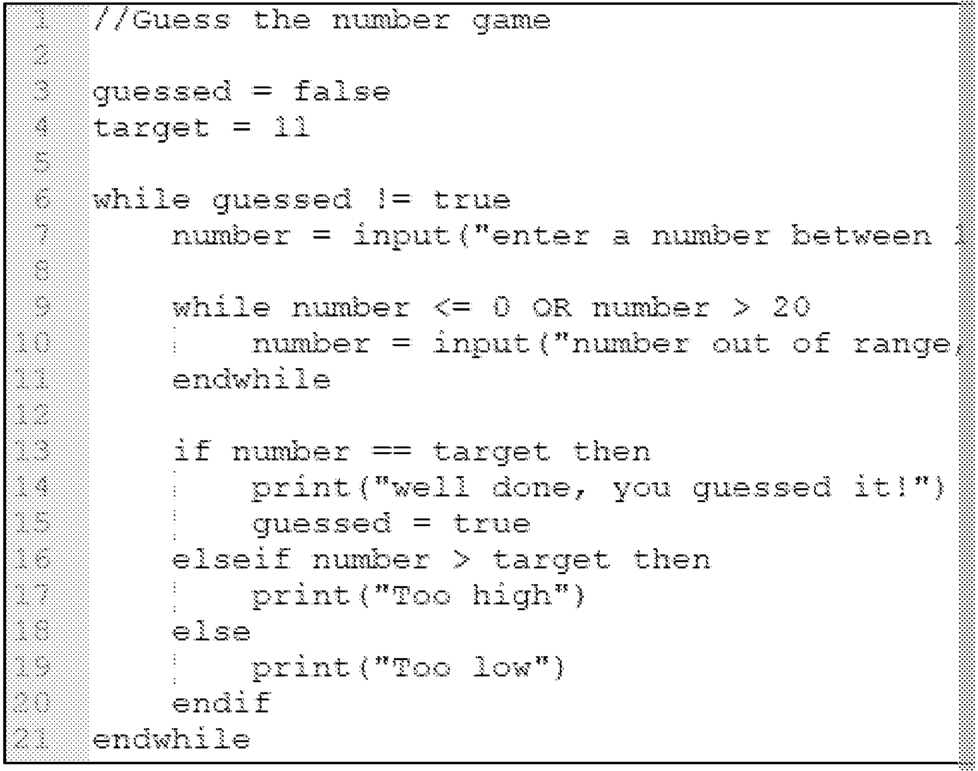


```
a = input("Enter first numbe
b = input("Enter second numb

c = a + b
if a < b then
    a = a + 1
    b = b - a
    c = a + b
    print(c)
else
    print(c)
endif
```

A	b	c	a < b	a
5	7	12	True	6
15	4			
17	19			
62	49			
23	11			

EXERCISE 10: IDENTIFY THE CONST

Study the example OCR Exam Reference Language carefully and complete the ta
algorithm show examples of sequence, selection and iteration.



```
1  //Guess the number game
2
3  guessed = false
4  target = 11
5
6  while guessed != true
7      number = input("enter a number between 1
8
9      while number <= 0 OR number > 20
10         number = input("number out of range,
11     endwhile
12
13     if number == target then
14         print("well done, you guessed it!")
15         guessed = true
16     elseif number > target then
17         print("Too high")
18     else
19         print("Too low")
20     endif
21 endwhile
```

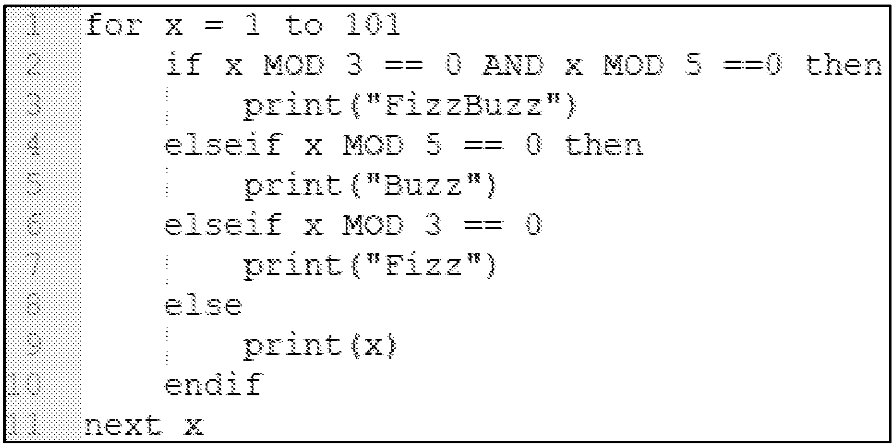
LINE NUMBER(S)	WHICH CONSTRUCT?	EXPLAI
3 and 4	*Sequence*	

EXERCISE 11: FIZZBUZZ TRAC

Complete a trace table for each of the two versions of the FizzBuzz maths game b

Explain which version is better, and why.

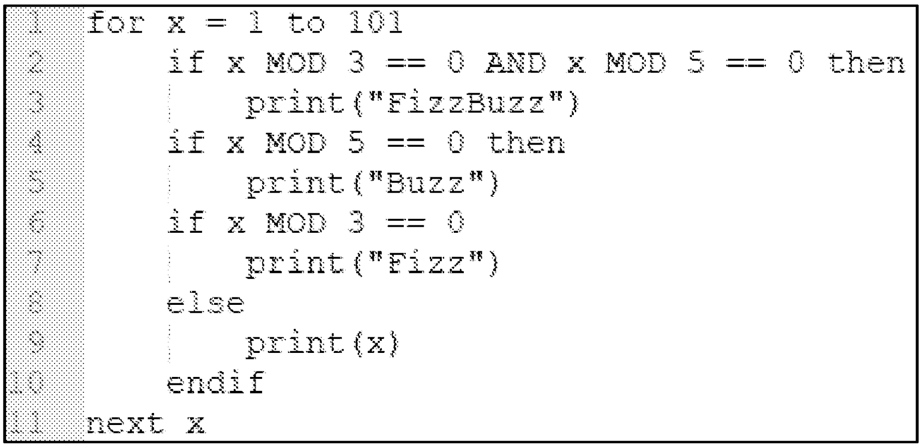
Version 1:



```
for x = 1 to 101
    if x MOD 3 == 0 AND x MOD 5 ==0 then
        print("FizzBuzz")
    elseif x MOD 5 == 0 then
        print("Buzz")
    elseif x MOD 3 == 0
        print("Fizz")
    else
        print(x)
    endif
next x
```

x	X MOD 3 = 0 AND x MOD 5 = 0	X MOD 5 = 0	X MO
9	*False*	*False*	*Tr*
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			

Version 2:



```
for x = 1 to 101
    if x MOD 3 == 0 AND x MOD 5 == 0 then
        print("FizzBuzz")
    if x MOD 5 == 0 then
        print("Buzz")
    if x MOD 3 == 0
        print("Fizz")
    else
        print(x)
    endif
next x
```

X	X MOD 3 = 0 AND x MOD 5 = 0	X MOD 5 = 0	X MOD
9	*False*	*False*	*Tr*
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			

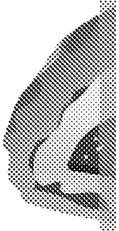
Which version is better and why?

EXERCISE 12: DIAL A PIZZA

Dial a Pizza wants a system that is easy to follow to make sure all the right quest completed and the correct waiting time is given to the customer, based on their

A pizza order is not **complete** until the following questions have been answered:

- Customer address recorded
- Thin, thick or stuffed crust base recorded
- Vegetarian or meat recorded
- Waiting time advised



The times for cooking pizzas are:

- Thin – 10 minutes
- Thick – 15 minutes
- Stuffed crust – 18 minutes

In this exercise you need to create your algorithm using a flow chart (on a separa correct symbols and arrows.

You will need to think about using ‘flag’ variables and your answer should use all

✂ -

EXERCISE 12: DIAL A PIZZA

Dial a Pizza wants a system that is easy to follow to make sure all the right quest completed and the correct waiting time is given to the customer, based on their

A pizza order is not **complete** until the following questions have been answered:

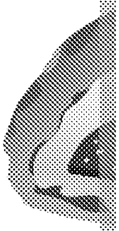
- Customer address recorded
- Thin, thick or stuffed crust base recorded
- Vegetarian or meat recorded
- Waiting time advised

The times for cooking pizzas are:

- Thin – 10 minutes
- Thick – 15 minutes
- Stuffed crust – 18 minutes

In this exercise you need to create your algorithm using a flow chart (on a separa correct symbols and arrows.

You will need to think about using ‘flag’ variables and your answer should use all



EXERCISE 13: COUNT UNTIL ZE

Write an algorithm using OCR Exam Reference Language which uses sequence, s

The algorithm must continue to ask the user for a number and continue to add t is entered. The total of all the numbers entered (except the 0) must be output t were entered.

EXERCISE 14: CALCULATE FAR

Midcentral Metrolink has installed a new system for paying fares using a contac loaded with money. The tram fares are calculated as follows:

5 miles or less	£2.00
5–10 miles	£3.25
Above 10 miles	£4.75

When the card is swiped at the start of the journey the tram station identity cod card reader device in the ticket booth. At the end of the journey, the card is swip exit barrier calculates the fare using a data structure called TramMatrix to find th tram stations and deducts the fare from the balance on the smartcard.

Passengers are offered discounts for off-peak travel:

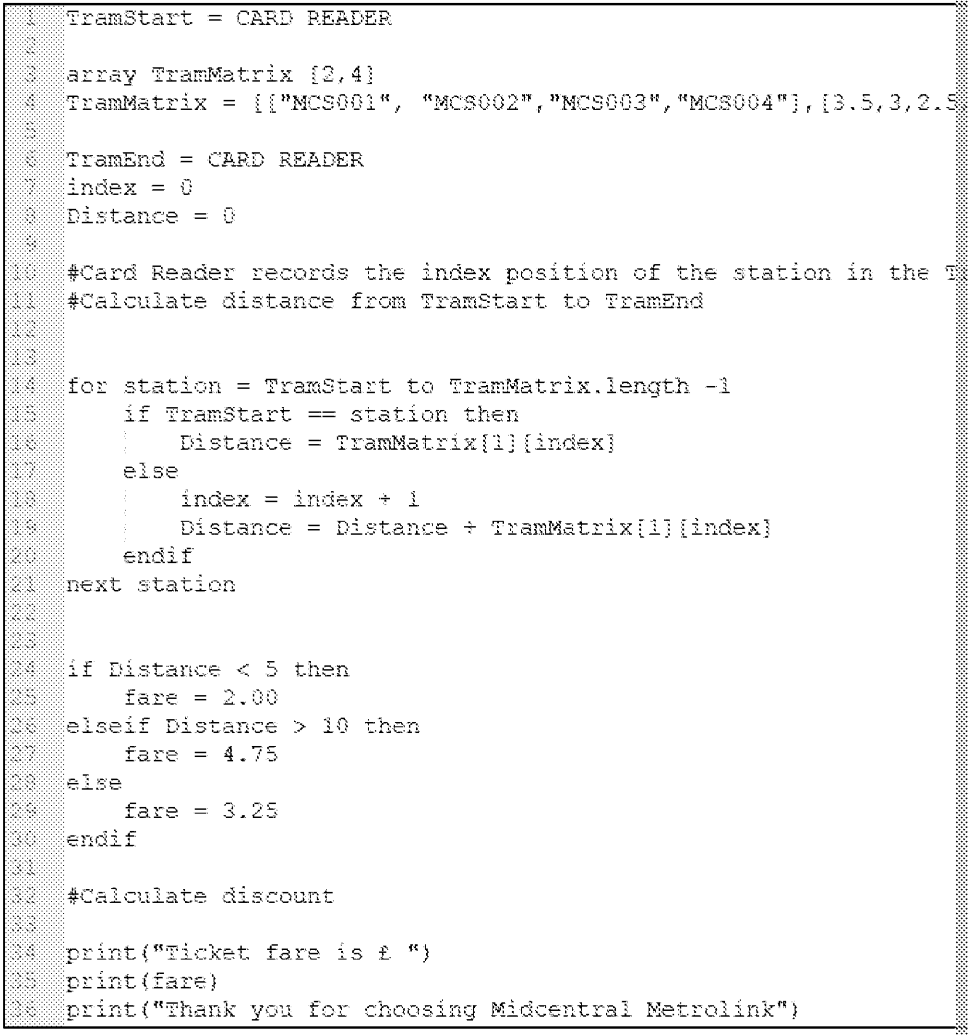
- 10% between 10am and 4pm Monday to Friday
- 15% all day Saturday and Sunday

An example of the TramMatrix is shown here:

STATIONID	DISTANCE (TO NEXT STATION)
MCS001	3.5
MCS002	3
MCS003	2.5
MCS004	4

Note: If the journey starts a StationID MCS002 the total

Study the OCR Exam Reference Language algorithm carefully and answer the qu



```
TramStart = CARD READER

array TramMatrix [2,4]
TramMatrix = [["MCS001", "MCS002","MCS003","MCS004"],[3.5,3,2.5

TramEnd = CARD READER
index = 0
Distance = 0

#Card Reader records the index position of the station in the T
#Calculate distance from TramStart to TramEnd


for station = TramStart to TramMatrix.length -1
    if TramStart == station then
        Distance = TramMatrix[1][index]
    else
        index = index + 1
        Distance = Distance + TramMatrix[1][index]
    endif
next station


if Distance < 5 then
    fare = 2.00
elseif Distance > 10 then
    fare = 4.75
else
    fare = 3.25
endif

#Calculate discount

print("Ticket fare is £ ")
print(fare)
print("Thank you for choosing Midcentral Metrolink")
```

Questions

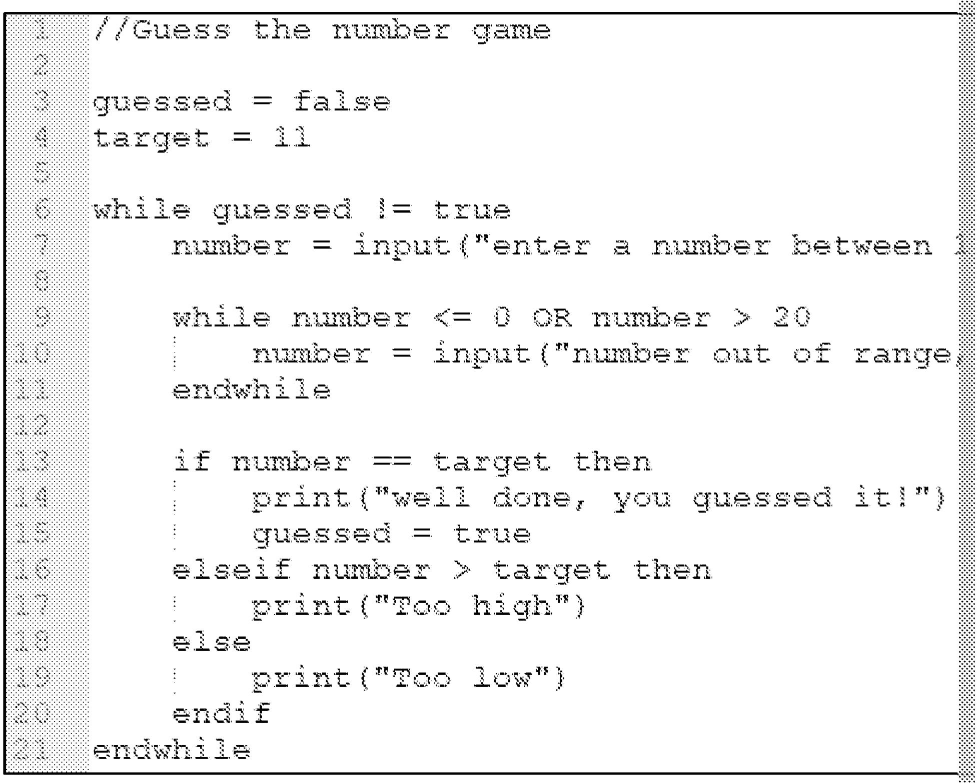
1. The algorithm currently continues adding up the distances instead of sto
   Identify the line where the error occurs and explain how to correct this.

2. The discount functionality has not yet been added. Write the OCR Exam
   the two discounts listed above.

   Hint: The variable name 'Time' may be useful in this answer.

EXERCISE 15: GUESSING GAME USING SU

The code below of a simple guessing game shows an example of nested iteratio sequence, selection and iteration can be combined.



```
//Guess the number game

guessed = false
target = 11

while guessed != true
    number = input("enter a number between 1

    while number <= 0 OR number > 20
        number = input("number out of range,
    endwhile

    if number == target then
        print("well done, you guessed it!")
        guessed = true
    elseif number > target then
        print("Too high")
    else
        print("Too low")
    endif
endwhile
```

On a separate piece of paper, re-write this algorithm using subprograms, to:

- allow a user to enter a new target number and return the target
- ask the user for their guess and return the guess

The target and the guess should be used as ‘parameters’ for the third subprogra outputs suitable messages.

Hint: You will need to call all three subprograms at least once.

EXERCISE 16: STRINGS AND SUBSTI

Write the following subprograms using OCR Exam Reference Language:

1. A subprogram which will ask for a string between 10 and 16 characters.
    a. The subprogram must check that a valid string has been entered and
    b. The string entered must be returned from the subprogram.
2. A subprogram that will accept the string (from your first subprogram) as a pa point and the end point for a substring.
    a. If the start or end point is not valid (because the string is not long en shown and the user asked again until a valid start or end point is ent
    b. The original string and the substring should then be printed with suit

Hint: You will need to check the length of the string in (1) and create a substring (from the For example, I might enter 'hashtagged' as myString and use SUBSTRING (4, 10, myString

Exercise 17: Area Tester

You are planning a program that will help younger students test their ability to co
rectangles and triangles.

1. The program must allow a user to choose whether they are testing them
2. The user must enter R to test rectangles, T to test triangles or X to exit.
3. The program must allow the student to enter the length and width for a r
   a triangle, and then enter their answer.
4. If the answer is incorrect, they have two more attempts before the corre
5. If the answer entered is correct, they can choose between rectangles or t
   program.

Your answer must use subprograms and be presented in a flow chart (on a separ

Exercise 17: Area Tester

You are planning a program that will help younger students test their ability to co
rectangles and triangles.

1. The program must allow a user to choose whether they are testing them
2. The user must enter R to test rectangles, T to test triangles or X to exit.
3. The program must allow the student to enter the length and width for a r
   a triangle, and then enter their answer.
4. If the answer is incorrect, they have two more attempts before the corre
5. If the answer entered is correct, they can choose between rectangles or t
   program.

Your answer must use subprograms and be presented in a flow chart (on a separ

Exercise 17: Area Tester

You are planning a program that will help younger students test their ability to co
rectangles and triangles.

1. The program must allow a user to choose whether they are testing them
2. The user must enter R to test rectangles, T to test triangles or X to exit.
3. The program must allow the student to enter the length and width for a r
   a triangle, and then enter their answer.
4. If the answer is incorrect, they have two more attempts before the corre
5. If the answer entered is correct, they can choose between rectangles or t
   program.

Your answer must use subprograms and be presented in a flow chart (on a separ

EXERCISE 18: PASSWORD CHECKER VA

In the 'Flow Charts and Subprograms' chapter there is an example of a simple pa

You now need to write a program that will allow the user to enter EITHER an inte must keep count of the number of integers and characters entered to ensure tha more characters in length AND contains three or more numbers 0–9.

Using OCR Exam Reference Language write separate subroutines to allow the use integers that make up the password and then check the password meets the crit integers. The program must then ask for the password to be entered again to che original.

Remember to correctly call your subroutines where appropriate.

Hint: Any subroutine can be used more than once in your main program.

EXERCISE 19: ENCRYPTION CIP

On a separate piece of paper, write an algorithm in OCR Exam Reference Langua
messages written in capital letters only. Your answer must use subprograms. Th

1. Ask for the message to be encrypted
2. Ask for a substitute number between 1 and 26
   a. produce an error message if this number is not in the correct range
   b. repeat until a suitable number is entered
3. Output the answer as a string, together with the original message

If any characters in the original message are not in capitals, then a question mark
encrypted string.

Hint: You will need to use concatenation in this exercise. How will you know a character i

EXERCISE 19: ENCRYPTION CIP

On a separate piece of paper, write an algorithm in OCR Exam Reference Langua
messages written in capital letters only. Your answer must use subprograms. Th

1. Ask for the message to be encrypted
2. Ask for a substitute number between 1 and 26
   a. produce an error message if this number is not in the correct range
   b. repeat until a suitable number is entered
3. Output the answer as a string, together with the original message

If any characters in the original message are not in capitals, then a question mark
encrypted string.

Hint: You will need to use concatenation in this exercise. How will you know a character i

EXERCISE 19: ENCRYPTION CIP

On a separate piece of paper, write an algorithm in OCR Exam Reference Langua
messages written in capital letters only. Your answer must use subprograms. Th

1. Ask for the message to be encrypted
2. Ask for a substitute number between 1 and 26
   a. produce an error message if this number is not in the correct range
   b. repeat until a suitable number is entered
3. Output the answer as a string, together with the original message

If any characters in the original message are not in capitals, then a question mark
encrypted string.

Hint: You will need to use concatenation in this exercise. How will you know a character i

EXERCISE 20: SIMPLE BATTLESH

In this exercise you will be creating the algorithm for a simple battleships game usi
arrays to store the position of ships, and a random number generator to choose yo

This should be written in OCR Exam Reference Language (on a separate piece of paper) and use subprograms.

1. Create a 2D array of 5 rows × 5 zeros, e.g.
   `row 1 = [0, 0, 0, 0, 0].`

2. Create arrays with the locations for your ships.
   a. Cruisers need 4 squares on the grid – you have one cruisers
   b. Submarines need 3 squares – you have two submarines
   c. Destroyers need 2 squares – you have two destroyers

   Example:
   `cruiser = [[0, 0], [0, 1], [0, 2], [0, 3]]`

3. Your algorithm must randomly calculate which element (row) to look at AND which index (column) in each element.

4. Each time a correct location is found, the algorithm must output a messa

5. The game should run for 10 attempts and then print out how many hits w

Hint: Nested loops will be helpful in this exercise.

Exercise 20A: Battleships Extension
Extend the functionality of the simple game so that the same location containing
more than once. If the same location is hit again (after the first hit), then the algo
and not add to the hit count.

EXERCISE 21: RPG GAME INVENT

Role-play games are very popular for all ages. They usually involve moving aroun solve puzzles or complete tasks to gain more items to store in an inventory. In or tasks, the player may need to use an item from their inventory.

You need to write an algorithm that will allow players to:

- View the contents of their inventory
- Add items to it
- Use items, i.e. delete them
- Exit from the inventory menu

On a separate piece(s) of paper:

1. Decompose the problem into tasks that can be solved.
2. Write suitable OCR Exam Reference Language subprograms to solve the prob

EXERCISE 21: RPG GAME INVENT

Role-play games are very popular for all ages. They usually involve moving aroun solve puzzles or complete tasks to gain more items to store in an inventory. In or tasks, the player may need to use an item from their inventory.

You need to write an algorithm that will allow players to:

- View the contents of their inventory
- Add items to it
- Use items, i.e. delete them
- Exit from the inventory menu

On a separate piece(s) of paper:

1. Decompose the problem into tasks that can be solved.
2. Write suitable OCR Exam Reference Language subprograms to solve the prob

EXERCISE 21: RPG GAME INVENT

Role-play games are very popular for all ages. They usually involve moving aroun solve puzzles or complete tasks to gain more items to store in an inventory. In or tasks, the player may need to use an item from their inventory.

You need to write an algorithm that will allow players to:

- View the contents of their inventory
- Add items to it
- Use items, i.e. delete them
- Exit from the inventory menu

On a separate piece(s) of paper:

1. Decompose the problem into tasks that can be solved.
2. Write suitable OCR Exam Reference Language subprograms to solve the prob

EXERCISE 22: MUSIC GIG

Up-and-coming band ***I Didn't Kno*** helicopter to play their gig at Lord holding a large music festival.

The safest place to land the helico of a lake which is connected to the bridge. They are due to perform a from the island to the stage. The b at one time and, unfortunately, no one torch.

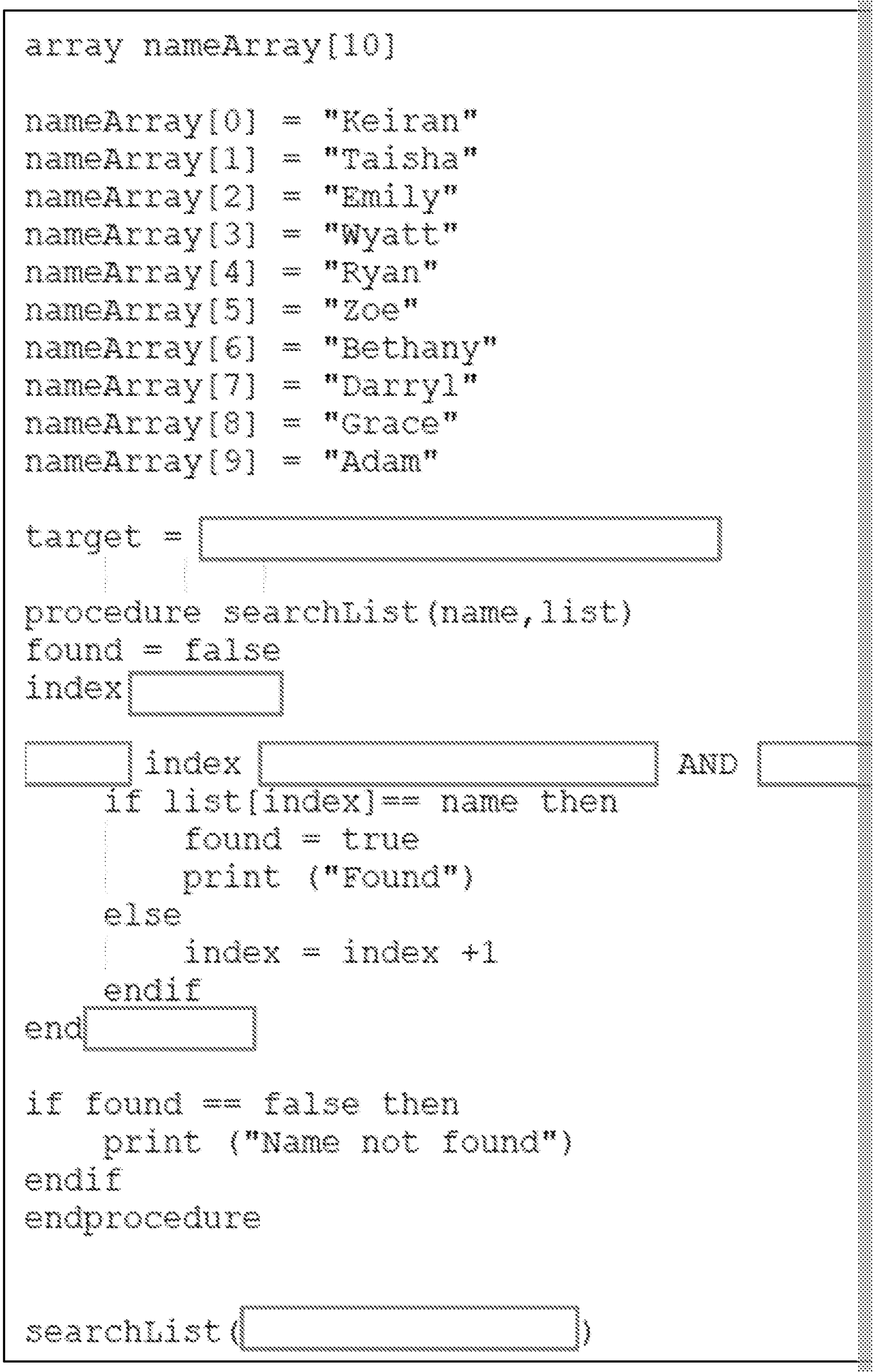
They are due on stage shortly and need to get everyone across to the stage as qu rush and the light is fading they must cross in the minimum time possible and m

The bridge is too long for the torch to be thrown back to the others; it must be c members have different fitness levels, which means they all cross at different sp minute, Bob in 2 minutes, Clair in 5 minutes and Danni in 8 minutes (as she sprai

Explain how you would solve this problem in the shortest possible time.

EXERCISE 23: FILL IN THE BLAN

Correct the linear search algorithm below so that it stops when the item has bee
Complete the blank spaces and check that the algorithm will run correctly when

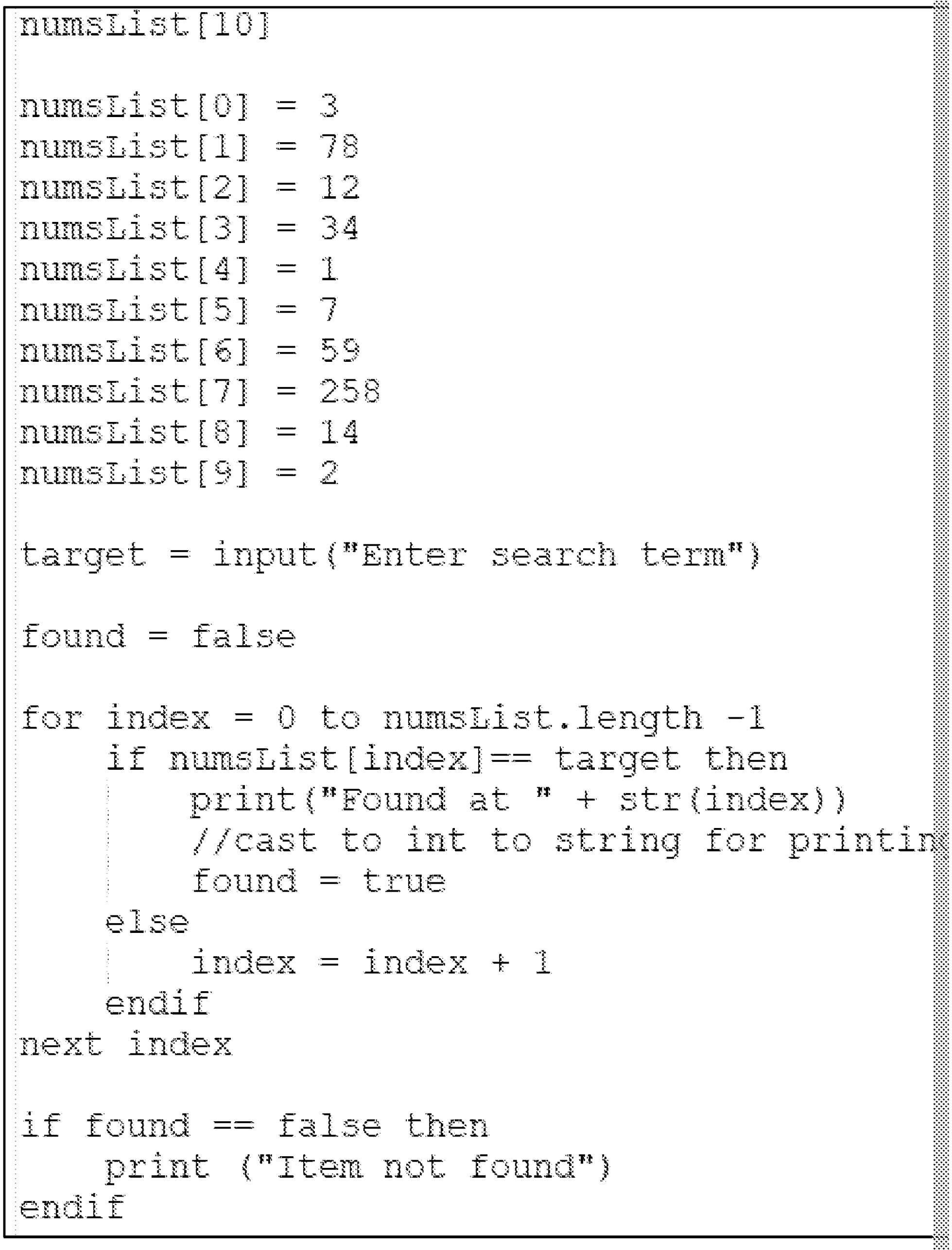


```
array nameArray[10]

nameArray[0] = "Keiran"
nameArray[1] = "Taisha"
nameArray[2] = "Emily"
nameArray[3] = "Wyatt"
nameArray[4] = "Ryan"
nameArray[5] = "Zoe"
nameArray[6] = "Bethany"
nameArray[7] = "Darryl"
nameArray[8] = "Grace"
nameArray[9] = "Adam"

target = □

procedure searchList(name,list)
found = false
index□

□ index □ AND □
    if list[index]== name then
        found = true
        print ("Found")
    else
        index = index +1
    endif
end□

if found == false then
    print ("Name not found")
endif
endprocedure


searchList(□)
```

EXERCISE 24: LINEAR SEARCHES AND TR

Complete the trace table exercises for these linear searches:

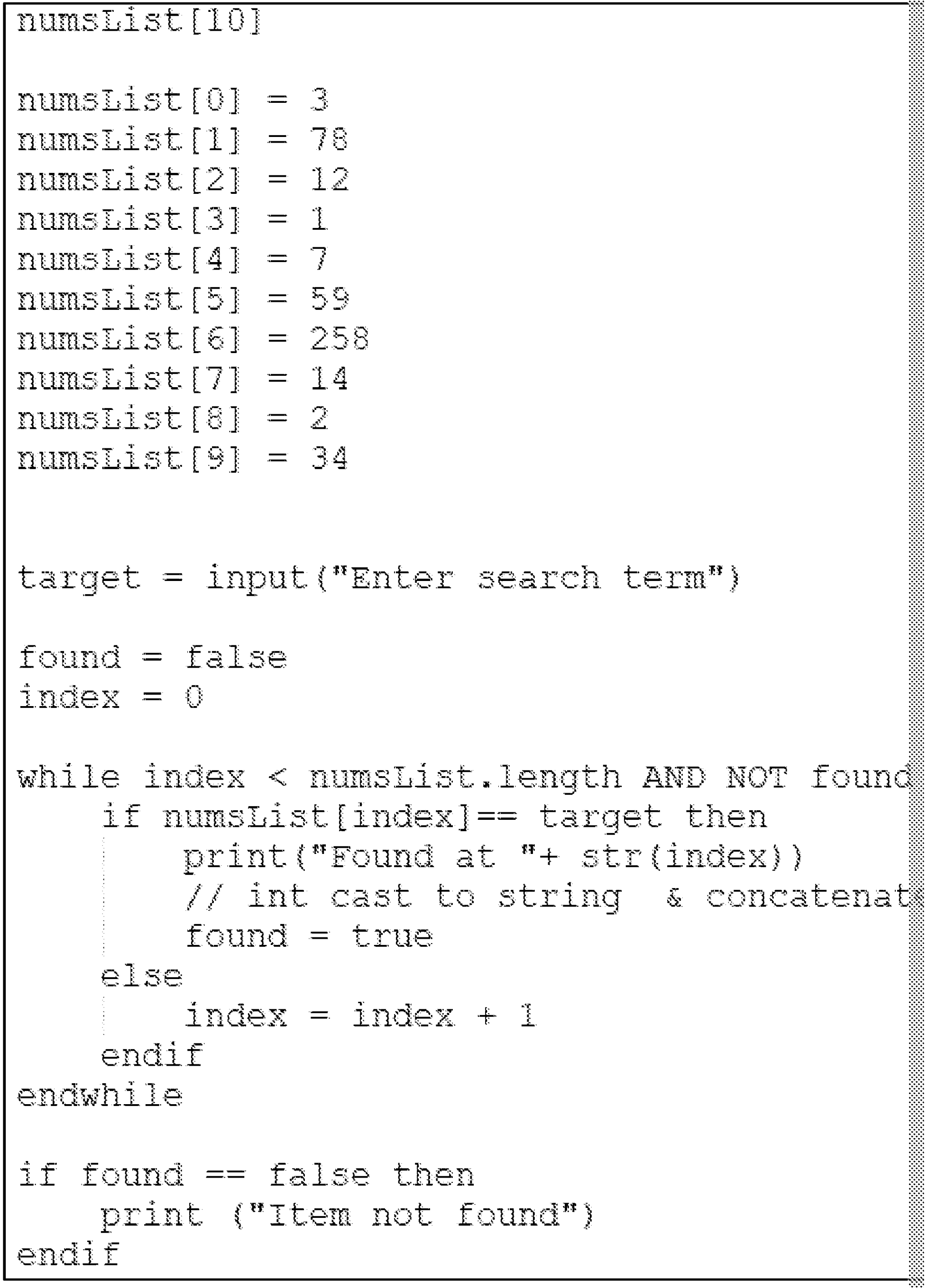
Linear search 1:



```
numsList[10]

numsList[0] = 3
numsList[1] = 78
numsList[2] = 12
numsList[3] = 34
numsList[4] = 1
numsList[5] = 7
numsList[6] = 59
numsList[7] = 258
numsList[8] = 14
numsList[9] = 2

target = input("Enter search term")

found = false

for index = 0 to numsList.length -1
    if numsList[index]== target then
        print("Found at " + str(index))
        //cast to int to string for printin
        found = true
    else
        index = index + 1
    endif
next index

if found == false then
    print ("Item not found")
endif
```

index	found	Target
0	False	34

Linear search 2:



```
numsList[10]

numsList[0] = 3
numsList[1] = 78
numsList[2] = 12
numsList[3] = 1
numsList[4] = 7
numsList[5] = 59
numsList[6] = 258
numsList[7] = 14
numsList[8] = 2
numsList[9] = 34


target = input("Enter search term")

found = false
index = 0

while index < numsList.length AND NOT found
    if numsList[index]== target then
        print("Found at "+ str(index))
        // int cast to string  & concatenat
        found = true
    else
        index = index + 1
    endif
endwhile

if found == false then
    print ("Item not found")
endif
```

index	found	target
0	False	1

Now explain which is most efficient and why, referring to the OCR Exam Referen explanation.

EXERCISE 25: BUBBLE SORT EXER

1. Complete the bubble sort for this array: [5, 1, 6, 2, 4, 3].

5	1	6	2	4	3

2. Complete this explanation of how to perform a bubble sort.
Hint: Remember that this sorting algorithm uses ITERATION.

1. Compare the first two elements in the array

2. Is the first element bigger than the second element?

3.

4.

5.

6.

EXERCISE 26: PUT THE BUBBLE SORT FLOW C

Complete the flow chart by writing the correct letter in the empty spaces.

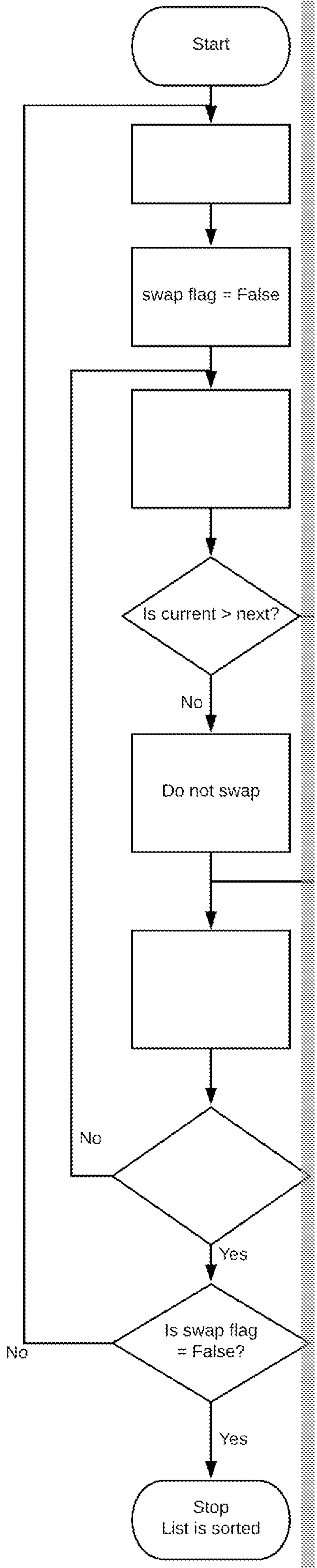
A Move one element along and set this as current element

B Has the last element in array been reached?

C Compare current element with next element

D Look at first element in array

E Swap the two elements



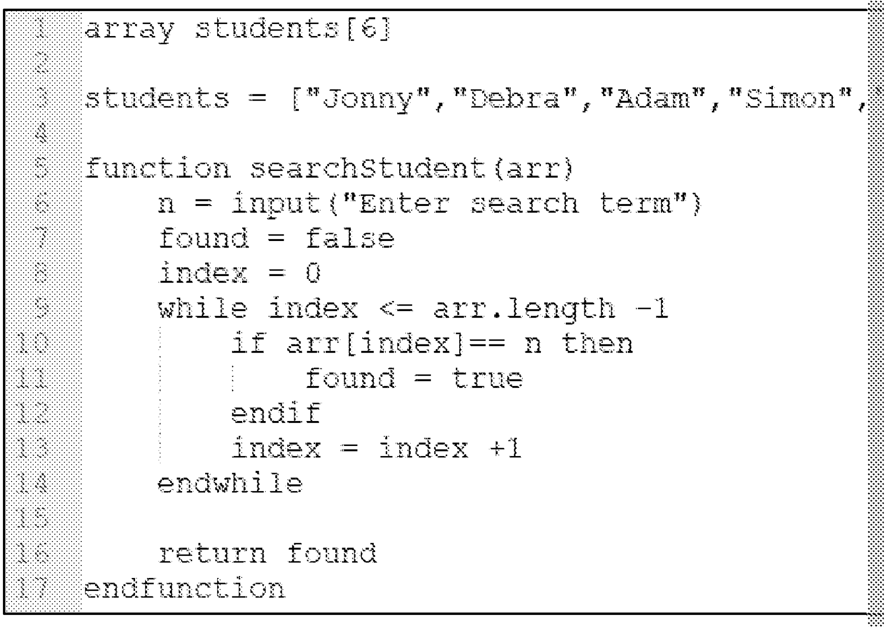
EXERCISE 27: SORTING AND SEARC

1. Complete these data sorts using a merge sort, ensuring that you show al

   a. 67,23,52,6,15,43,11,3

   b. 92,24,2,28,1,7,13,12

2. Samira is writing a simple program to allow a user to enter a name to be OCR Exam Reference Language for the algorithm she wants to use.



```
1  array students[6]
2
3  students = ["Jonny","Debra","Adam","Simon",
4
5  function searchStudent(arr)
6      n = input("Enter search term")
7      found = false
8      index = 0
9      while index <= arr.length -1
10         if arr[index]== n then
11             found = true
12         endif
13         index = index +1
14     endwhile
15
16     return found
17 endfunction
```

   a. What type of search is being used?

   b. Describe the algorithm, in terms of its inputs and outputs. What does

   c. The algorithm could be amended to be more efficient. State which lin changed and explain how the change will make the algorithm more e

3. Explain how the bubble sort will work to sort this simple array from:

22	4	13	9	17	1

to

1	4	9	13	17	22

The array will start at index position [0].

4. There are two different measurements for the efficiency of an algorithm
Discuss the merge sort and the bubble sort in terms of their time and sp

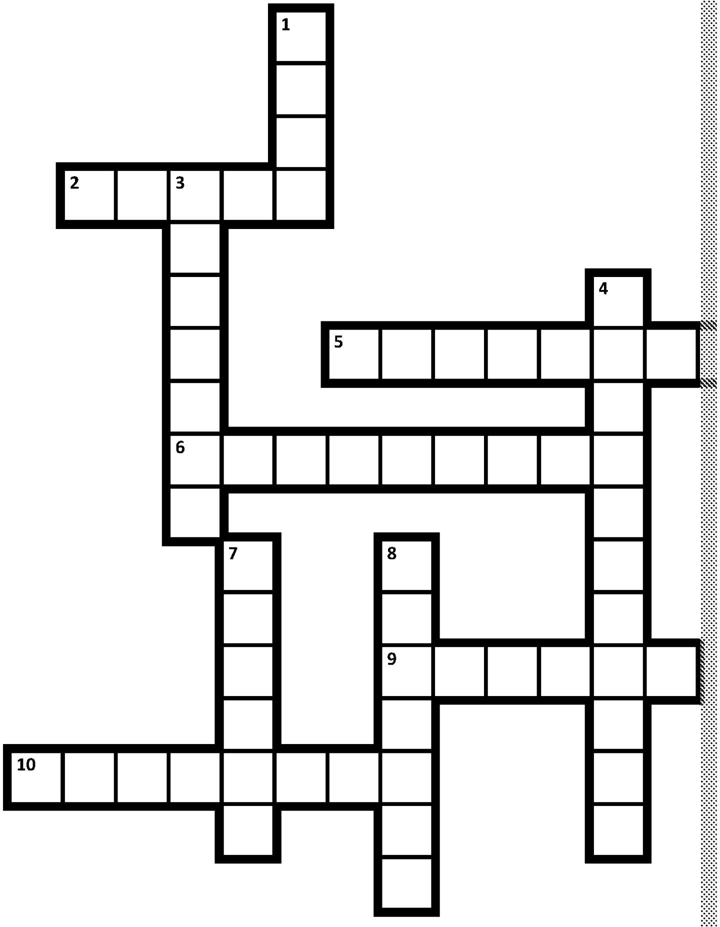
5. Describe this subprogram in terms of its inputs and outputs. What does

```
array nums[8]

nums = [15,63,14,89,12,3,62,51]

function FindSmallest(arr)
    smallest = arr[0]
    for i = 0 to arr.length-1
        if arr[i] < smallest then
            smallest = arr[i]
        endif

    return smallest
endfunction
```

6. Jack has been given homework to write an algorithm to search a variety
Which search method would be most suitable for use with this array, and

[2, 6, 9, 12, 23, 41, 76, 84, 92]

Crossword One



Across

2 Something put into a process (5)

5 An ordered set of steps or instructions (8)

6 A series of instructions that solves a problem in a finite number of steps (9)

9 The result of processing (6)

10 A location in memory where data is stored (8)

Down

1 Something that is

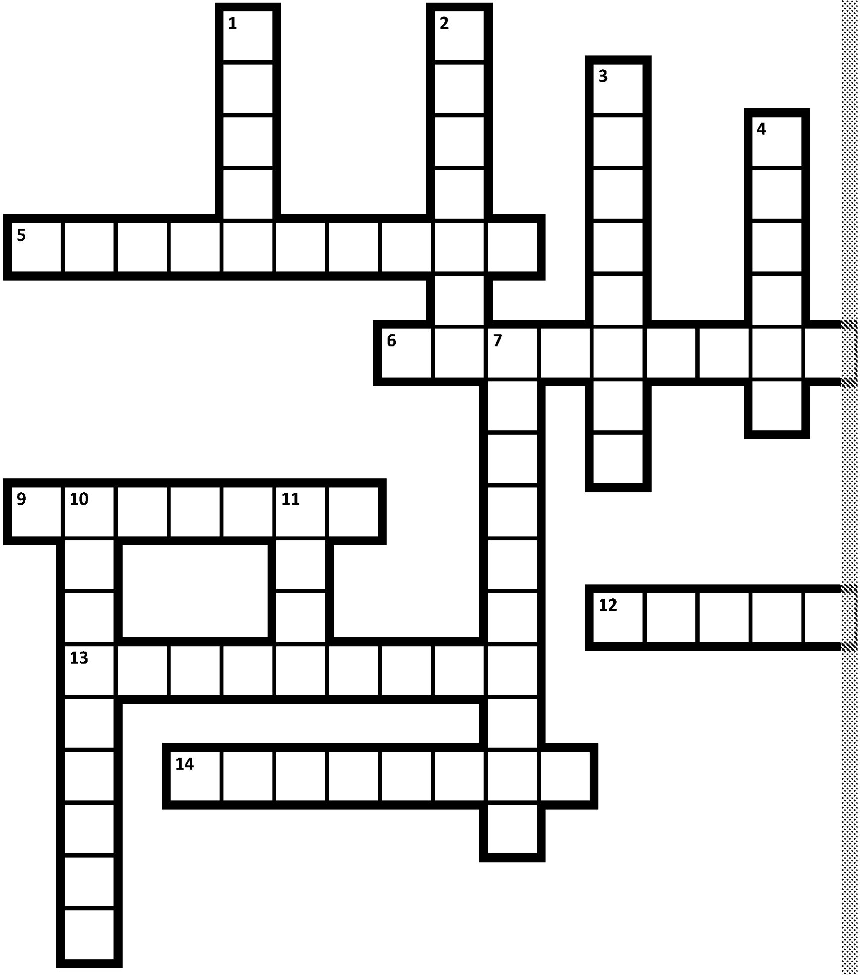
3 Code that tells a c algorithm (7)

4 Written in a way t completely clear (

7 A picture, piece of something (6)

8 A series of steps p (7)

Crossword Two



Across

5 This must be unique and meaningful (10)

6 Written in a way that makes it completely clear what is meant (11)

9 A series of steps performed to achieve a result (7)

12 The value stored here never changes when a program is run (8)

13 A series of instructions that solves a problem in a finite number of steps (9)

14 The result of integer division (8)

Down

1 Something put int

2 This data type can

3 This may change a

4 The result of proce

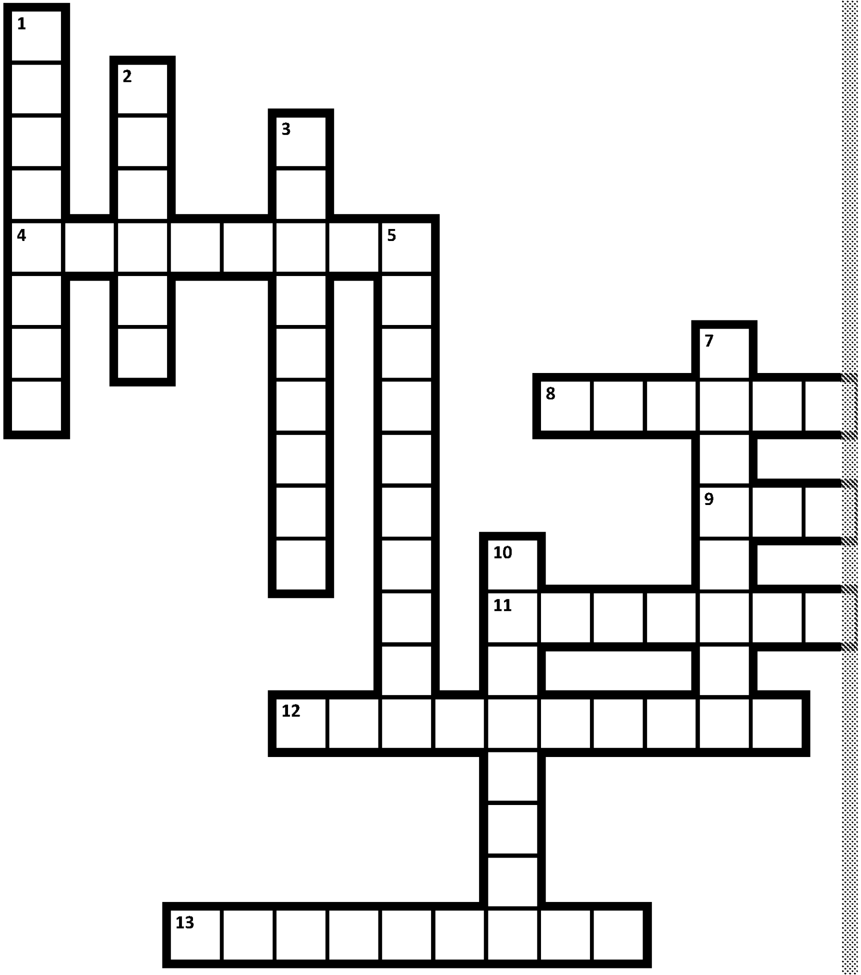
7 The term used to d location a value (1

8 An ordered set of

10 The result of using

11 The symbol for mu

Crossword Three



Across

4 A base integer is raised to the power of this integer (8)

8 The result of using the modulus operator (9)

9 This keyword gets a value into your algorithm from the keyboard (5)

11 Written in a way that makes it completely clear what is meant (11)

12 This must be unique and meaningful (10)

13 A series of instructions that solves a problem in a finite number of steps that always ends (9)

Down

1 An ordered set of

2 The result of proce

3 This means instruc program are repea

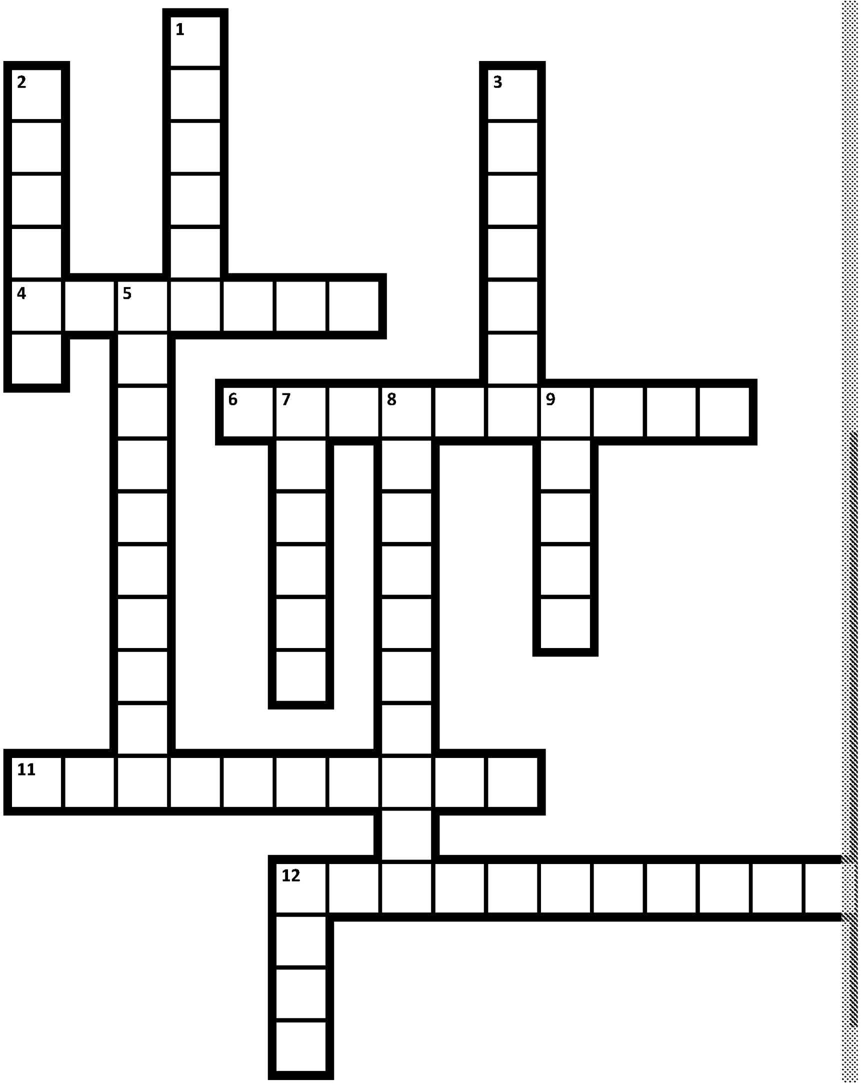
5 A method to test a no logic errors (5,5

6 This describes whe whether a conditio before taking actio

7 This may change a

10 The result of integ

CROSSWORD FOUR



Across

4 The term used to describe a programming construct, such as a loop, placed inside another programming construct (7)

6 A paper-based method for checking an algorithm (5,5)

11 The name given to a variable (10)

12 The process of joining two strings (13)

Down

1 The result of processing (6)

2 A sequence of characters surr quotation marks (6)

3 Used to describe each item in

5 Used to indicate the start of a (10)

7 This is a feature of a function

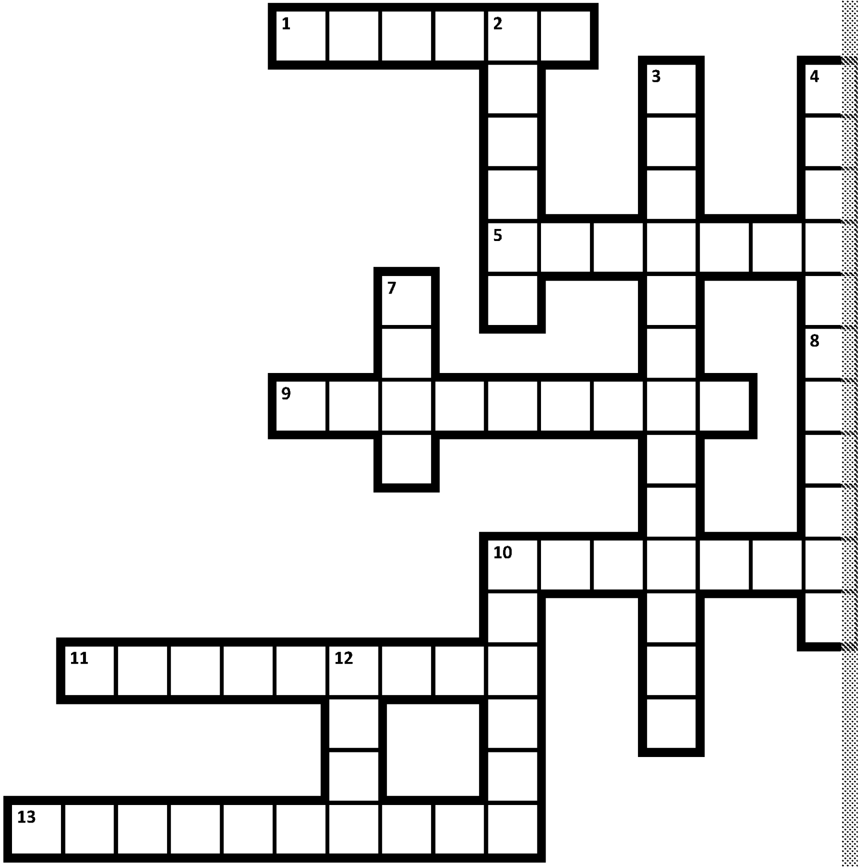
8 The process of changing, for e (10)

9 Data structure to store multip name (5)

10 Written in a way that makes i meant (11)

12 This is the term used to start a program (4)

Crossword Five



Across

1 This sorting method has the most efficient use of memory (6)

5 A series of instructions that solves a problem in a finite number of steps that always ends (9)

8 A data structure that can contain many items under one variable name (5)

9 This sort is very quick when adding items to a sorted array (9)

10 A problem-solving approach (5,5)

11 The term used to describe repeating a process in an algorithm (9)

13 The term used to describe how well an algorithm works (10)

Down

2 This algorithm loo

3 The process of bre smaller sub-proble

4 The term used to d unnecessary detai

6 This algorithm has amount of data inc

7 The process of an data structure (4)

10 A search method t sorted (6)

12 A measurement of (4)

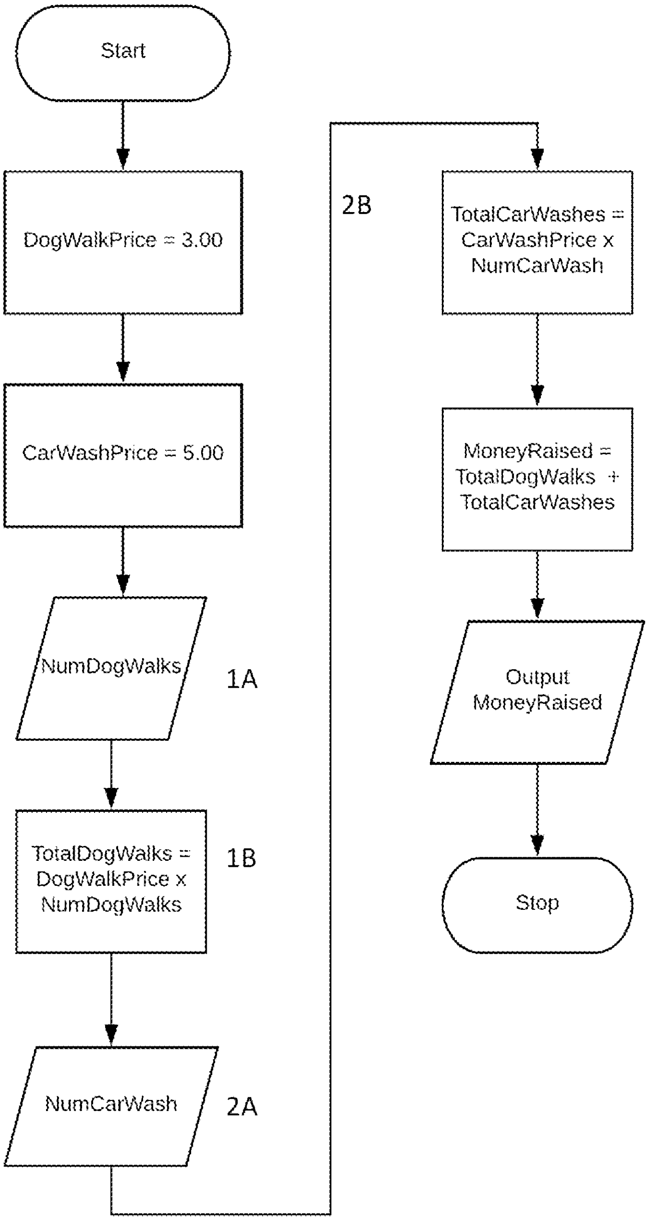
Suggested Answers

Exercises

Exercise 1

Inputs	Process
1. Number of dog walks *2. Number of car washes* *3. Price per dog walk* *4. Price per car wash*	*Total dog walks = No. of dog walks × Price per dog w* *Total car washes = No. of car washes × Price per ca* *Money raised = Total dog walks + Total car washes*

Exercise 2


Start
DogWalkPrice = 3.00
CarWashPrice = 5.00
NumDogWalks
1A
TotalDogWalks = DogWalkPrice x NumDogWalks
1B
NumCarWash
2A
2B
TotalCarWashes = CarWashPrice x NumCarWash
MoneyRaised = TotalDogWalks + TotalCarWashes
Output MoneyRaised
Stop

Shapes 1A a
interchange
and still pro
long as 1A c
comes befo

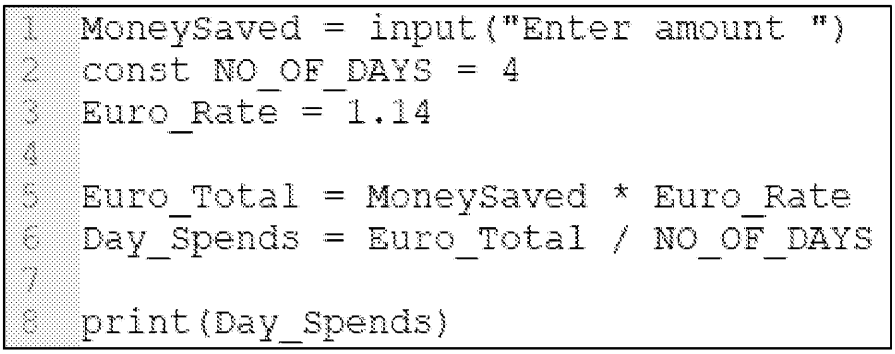
Exercise 3

Expression	Constant or Variable?	R
currentTemp = 30	Variable	The **identifier** says that this is t this could change when the alg
pi = 3.14159	Constant	The mathematical value of pi is
diameter = 34.5	Variable	The **identifier** gives a value for algorithm runs.
boilPoint = 100	Constant	The boiling point of water, at se
currentShoeSize = 5.5	Variable	The **identifier** gives a value for change as the algorithm runs.
daysInWeek = 7	Constant	The number of days in a week i
minsInHour = 60	Constant	The number of minutes in an ho
playerOneDiceRoll = 5	Variable	The **identifier** gives a value for algorithm runs.
gramToOunce = 0.0352	Constant	The number of grams to ounce
playerName = "Charlotte"	Variable	The **identifier** gives a value for the algorithm runs.

Exercise 4

INPUTS	PROCESS	OUTPUTS

MoneySaved
No_of_Days
Euro_rate
	Euro_Total = MoneySaved × Euro_rate
	Day_Spends = Euro_Total / No_of_Days
		Day_Spends



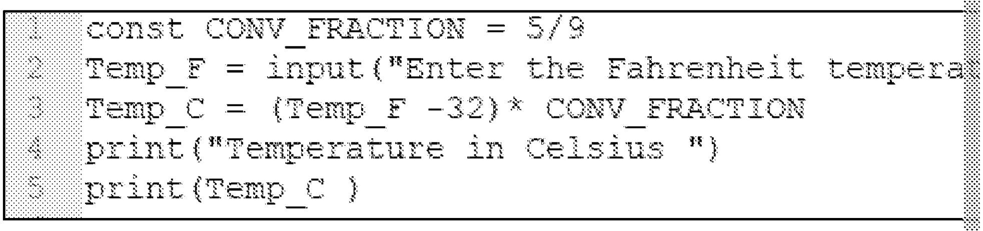
```
1 MoneySaved = input("Enter amount ")
2 const NO_OF_DAYS = 4
3 Euro_Rate = 1.14
4
5 Euro_Total = MoneySaved * Euro_Rate
6 Day_Spends = Euro_Total / NO_OF_DAYS
7
8 print(Day_Spends)
```

No
nar
Line
inp
the
Eur
inp

The
nun
eve
sho

Exercise 5

INPUTS	PROCESS	OUTPUTS
Temp_F Fraction	Temp_C = (Temp_F-32)*Fraction	Temp_C



```
1 const CONV_FRACTION = 5/9
2 Temp_F = input("Enter the Fahrenheit tempera
3 Temp_C = (Temp_F -32)* CONV_FRACTION
4 print("Temperature in Celsius ")
5 print(Temp_C )
```

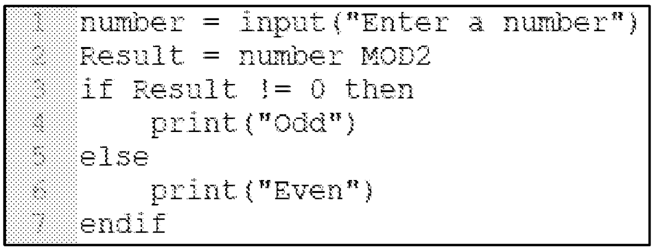
Note: The value of 32 could also be programmed as a constant in this example. The use o necessary in this example but it is good practice for any value that does not change.

Exercise 6

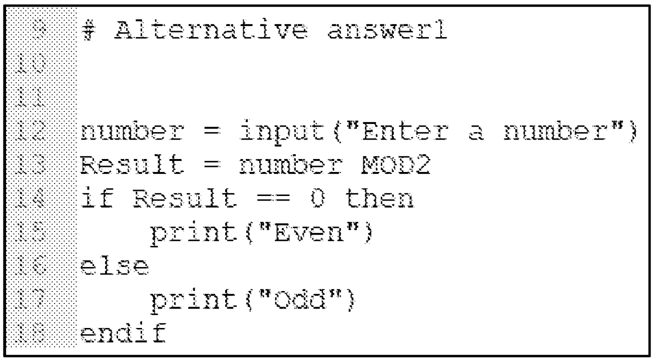
INPUTS	PROCESS	OUTPUTS
number	Result = number MOD 2 If Result ≠ 0 THEN Output Odd Else Output Even #Alternative Process 1 If Result = 0 THEN Output Even Else Output Odd #Alternative Process 2 If Result >0 THEN Output Odd Else Output Even	Odd or even

This should be written BOTH in OCR Exam Reference Language AND as a flow cha

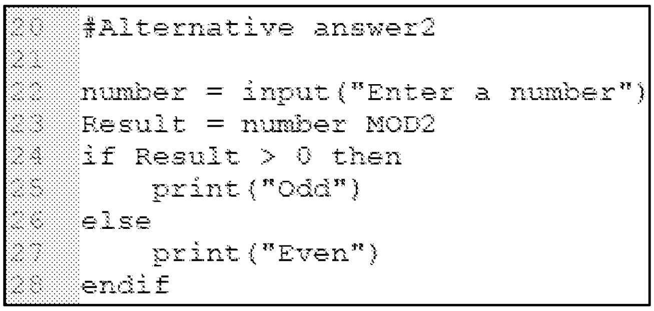
Pseudocode



```
number = input("Enter a number")
Result = number MOD2
if Result != 0 then
    print("Odd")
else
    print("Even")
endif
```

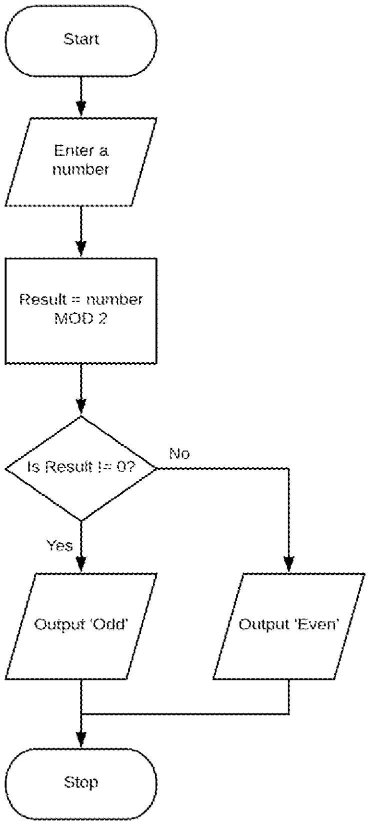


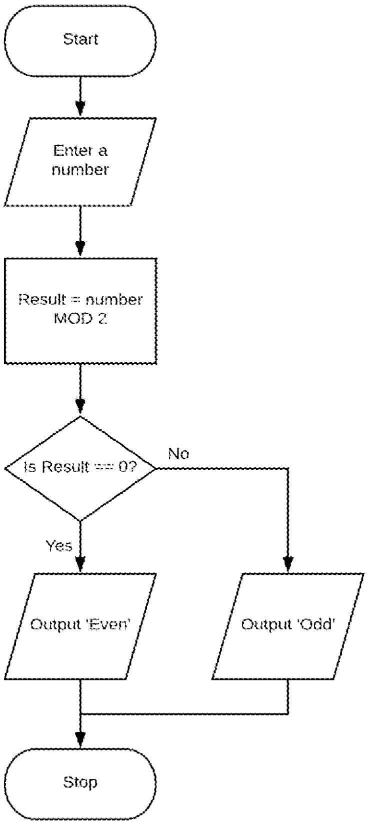
```
# Alternative answer1


number = input("Enter a number")
Result = number MOD2
if Result == 0 then
    print("Even")
else
    print("Odd")
endif
```



```
#Alternative answer2

number = input("Enter a number")
Result = number MOD2
if Result > 0 then
    print("Odd")
else
    print("Even")
endif
```

Flow charts



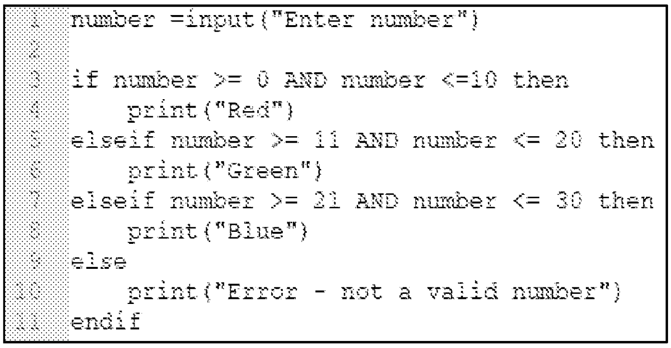


Exercise 7

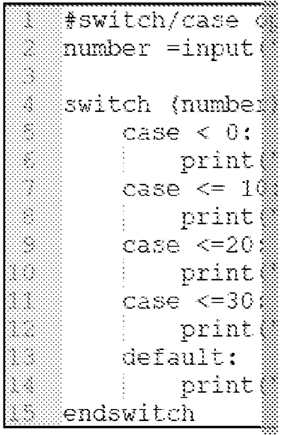
INPUTS	PROCESS	OUTPUTS
number	If number is between 0 and 10 then output red If number is between 11 and 20 then output green If number is between 21 and 30 then output blue	Red, green or blue Error – not a valid n

This should be written BOTH in OCR Exam Reference Language AND as a flow cha

Exam Reference Language



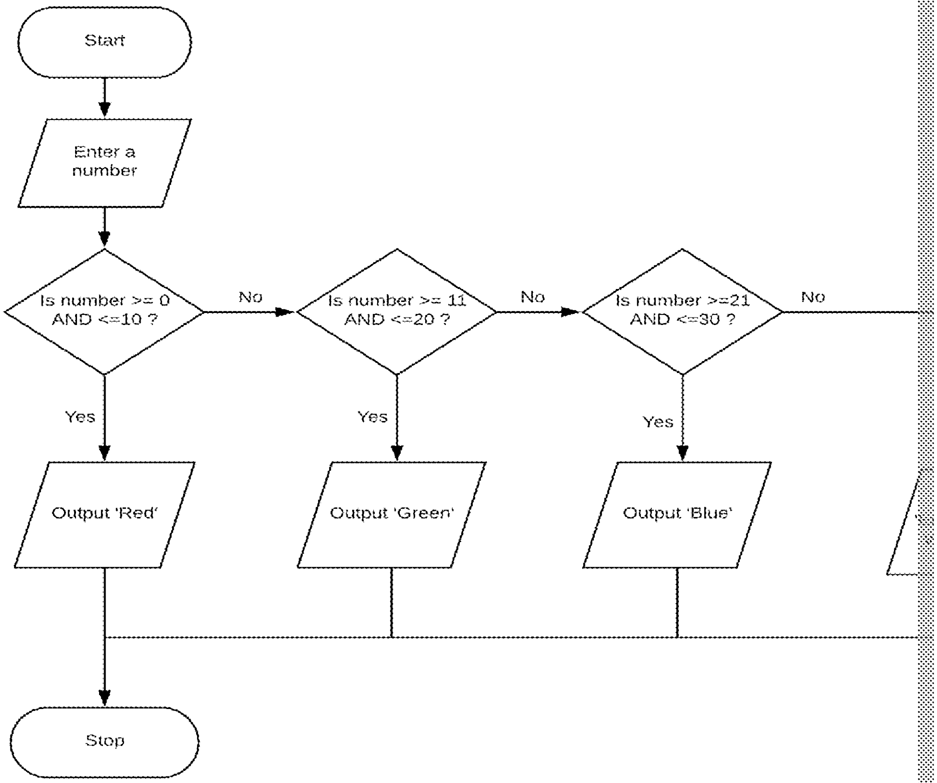
```
number =input("Enter number")

if number >= 0 AND number <=10 then
    print("Red")
elseif number >= 11 AND number <= 20 then
    print("Green")
elseif number >= 21 AND number <= 30 then
    print("Blue")
else
    print("Error - not a valid number")
endif
```



```
#switch/case
number =input

switch (number
    case < 0:
        print
    case <= 1
        print
    case <=20
        print
    case <=30
        print
    default:
        print
endswitch
```

Note: It is important that the greater than or equal to / less than or equal to symbol is us correct number range is tested.

Flow chart



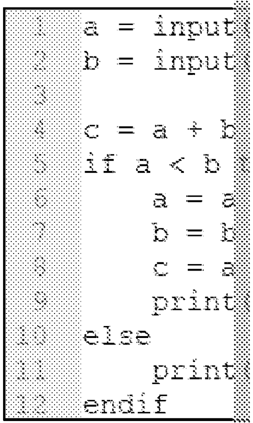
Exercise 8

num1	num2	num1 >= num2	num1
5	9	False	4
3	8	False	5
2	10	False	8
12	5	True	17
1	20	False	19
17	3	True	20

Exercise 9

a	b	c	a < b	a	b
5	7	12	True	6	1
15	4	19	False		
17	19	36	True	18	1
62	49	111	False		
23	11	34	False		

Note: Where values do not change (a,b) they do not need to be repeated in the trace table.



```
a = input
b = input

c = a + b
if a < b
    a = a
    b = b
    c = a
    print
else
    print
endif
```

Exercise 10

Lines numbers	Construct	Explanation
3 and 4	Sequence	The instructions follow one another in sequence.
6 to 21	Iteration	Line 6 shows a WHILE loop using condition-controlled iterat
14 and 15	Sequence	The instructions follow one another in sequence.
9 to 11	Iteration	This shows another WHILE loop 'nested' inside the main WH of condition-controlled iteration as it only stops when the n 20.
13 to 20	Selection	This is an ELSEIF statement with three possible options. It co between Lines 6 and 21. When the number entered equals 'guessed' is set to True and the condition for the main WHIL True.

Note: 'Nesting' means combining code together. In this example, an inner WHILE loop is outer WHILE loop between Lines 6 and 21.

Exercise 11

Version 1

x	X MOD 3 == 0 AND x MOD 5 == 0	X MOD 5 == 0	X MOI
9	*False*	*False*	*True*
10	False	True	False
11	False	False	False
12	False	False	True
13	False	False	False
14	False	False	False
15	True	True	True
16	False	False	False
17	False	False	False
18	False	False	True
19	False	False	False
20	False	True	False

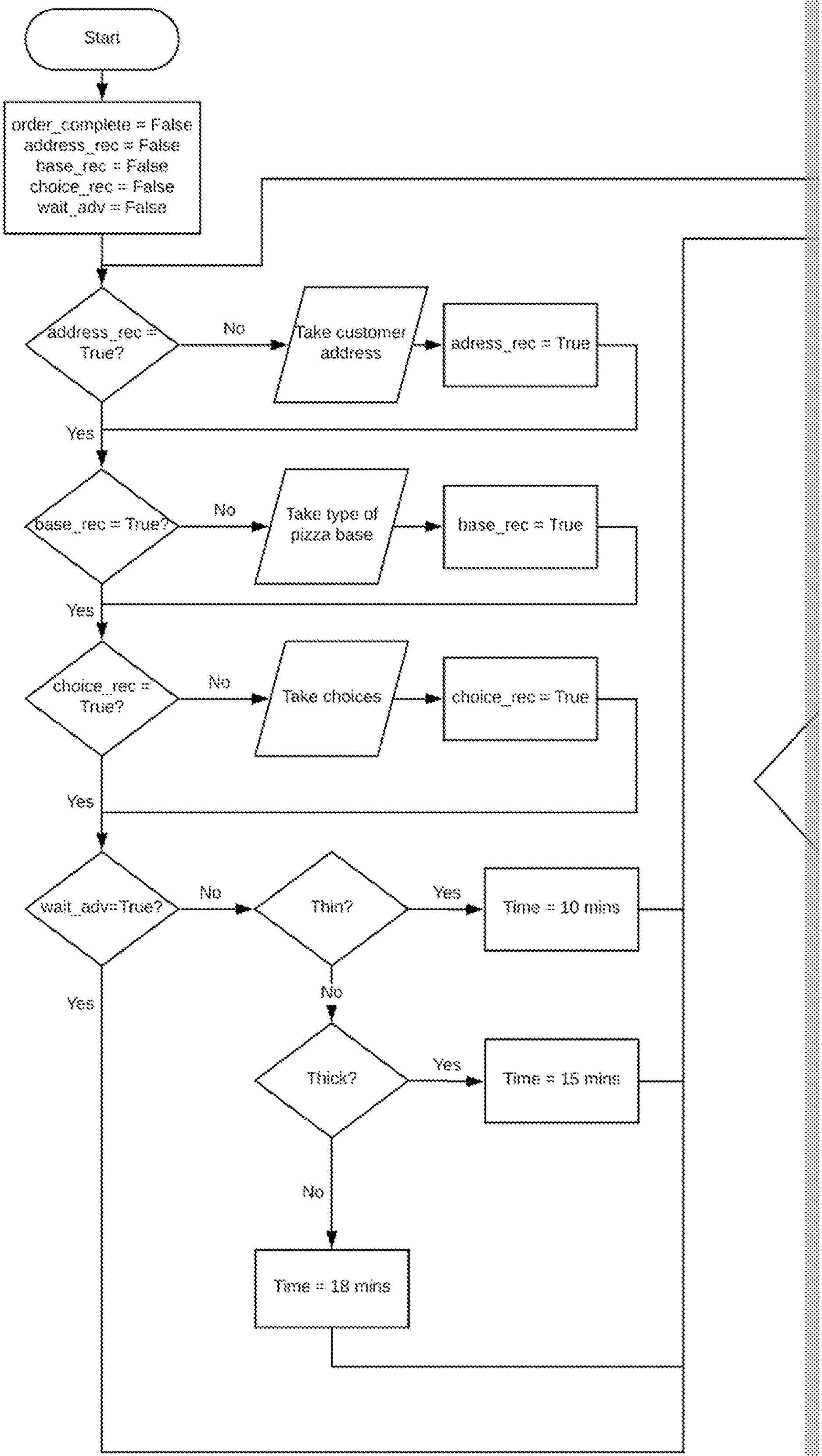
Version 2

x	X MOD 3 == 0 AND x MOD 5 == 0	X MOD 5 == 0	X MOI
9	*False*	*False*	*True*
10	False	True	False
11	False	False	False
12	False	False	True
13	False	False	False
14	False	False	False
15	True	True	True
16	False	False	False
17	False	False	False
18	False	False	True
19	False	False	False
20	False	True	False

Explain which version is better, and why

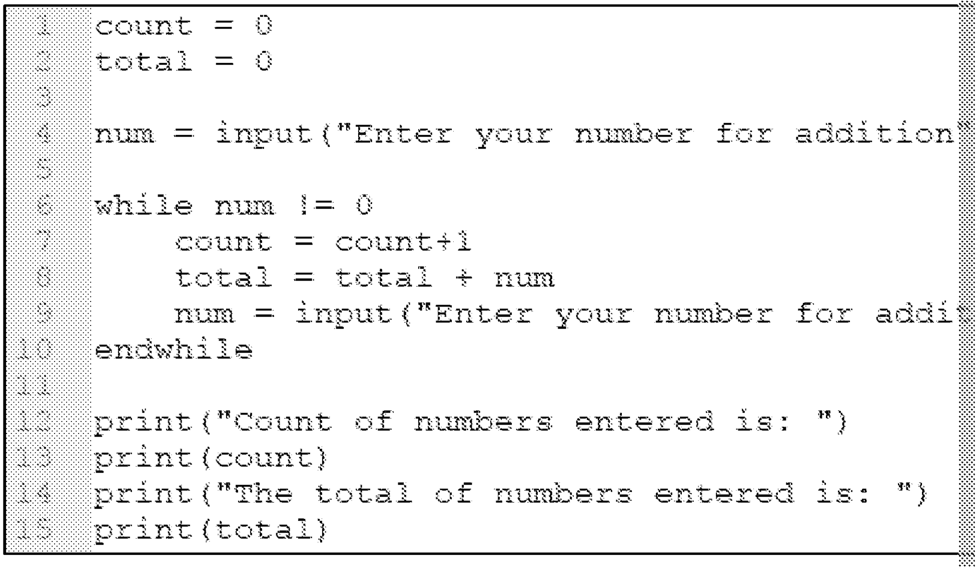
Version 2 does not work correctly, as you can see in the Trace Table for numbers 10, 1 linked and, instead of testing whether the number meets one of the three possible cond separately, leading to these multiple outputs. Version 1 is the correct version for these re

Exercise 12



There are five 'flags' set at the start of the process; each of the conditions is checked and
the appropriate flag to True. There is a final check at the end of the algorithm; if all four
the order is complete and the process finishes.

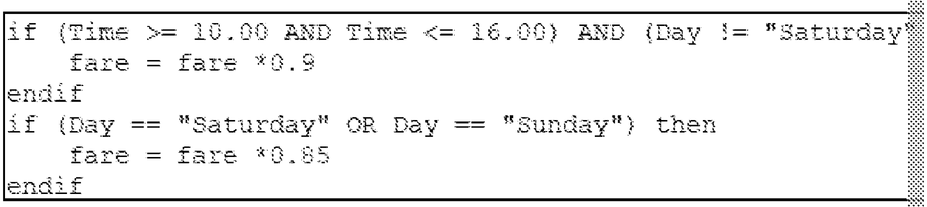
Exercise 13



```
count = 0
total = 0

num = input("Enter your number for addition"

while num != 0
    count = count+1
    total = total + num
    num = input("Enter your number for addi
endwhile

print("Count of numbers entered is: ")
print(count)
print("The total of numbers entered is: ")
print(total)
```

Exercise 14

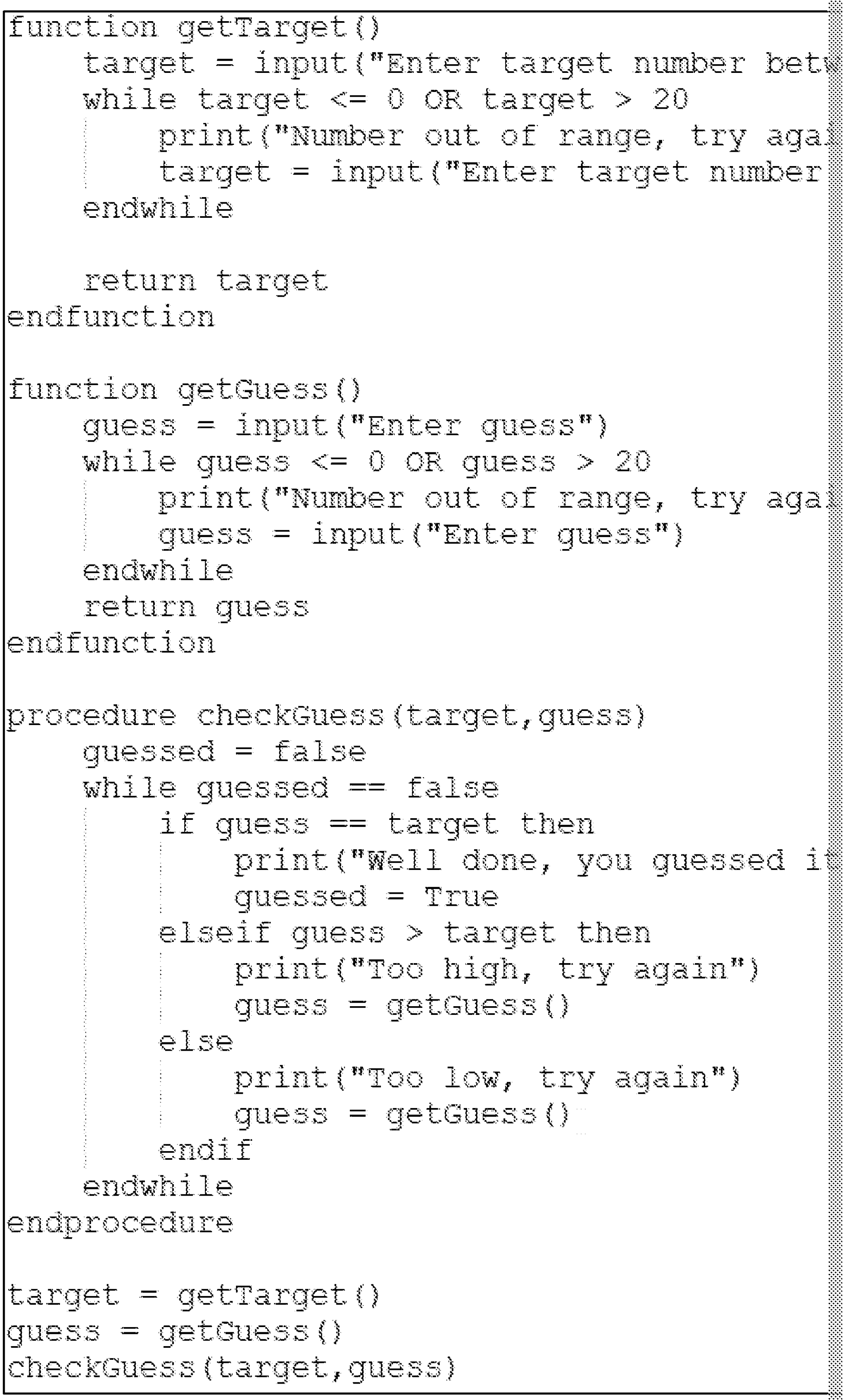
1. The error is on Line 14 as the FOR loop runs to the end of the TramMatrix. This line
   `TramStart to TramEnd`

2. 

```
if (Time >= 10.00 AND Time <= 16.00) AND (Day != "Saturday"
    fare = fare *0.9
endif
if (Day == "Saturday" OR Day == "Sunday") then
    fare = fare *0.85
endif
```

   Note: This could also be written using an IF/ELSEIF statement to combine the two IF

Exercise 15



```
function getTarget()
    target = input("Enter target number betw
    while target <= 0 OR target > 20
        print("Number out of range, try agai
        target = input("Enter target number
    endwhile

    return target
endfunction

function getGuess()
    guess = input("Enter guess")
    while guess <= 0 OR guess > 20
        print("Number out of range, try agai
        guess = input("Enter guess")
    endwhile
    return guess
endfunction

procedure checkGuess(target,guess)
    guessed = false
    while guessed == false
        if guess == target then
            print("Well done, you guessed it
            guessed = True
        elseif guess > target then
            print("Too high, try again")
            guess = getGuess()
        else
            print("Too low, try again")
            guess = getGuess()
        endif
    endwhile
endprocedure

target = getTarget()
guess = getGuess()
checkGuess(target,guess)
```

Exercise 16

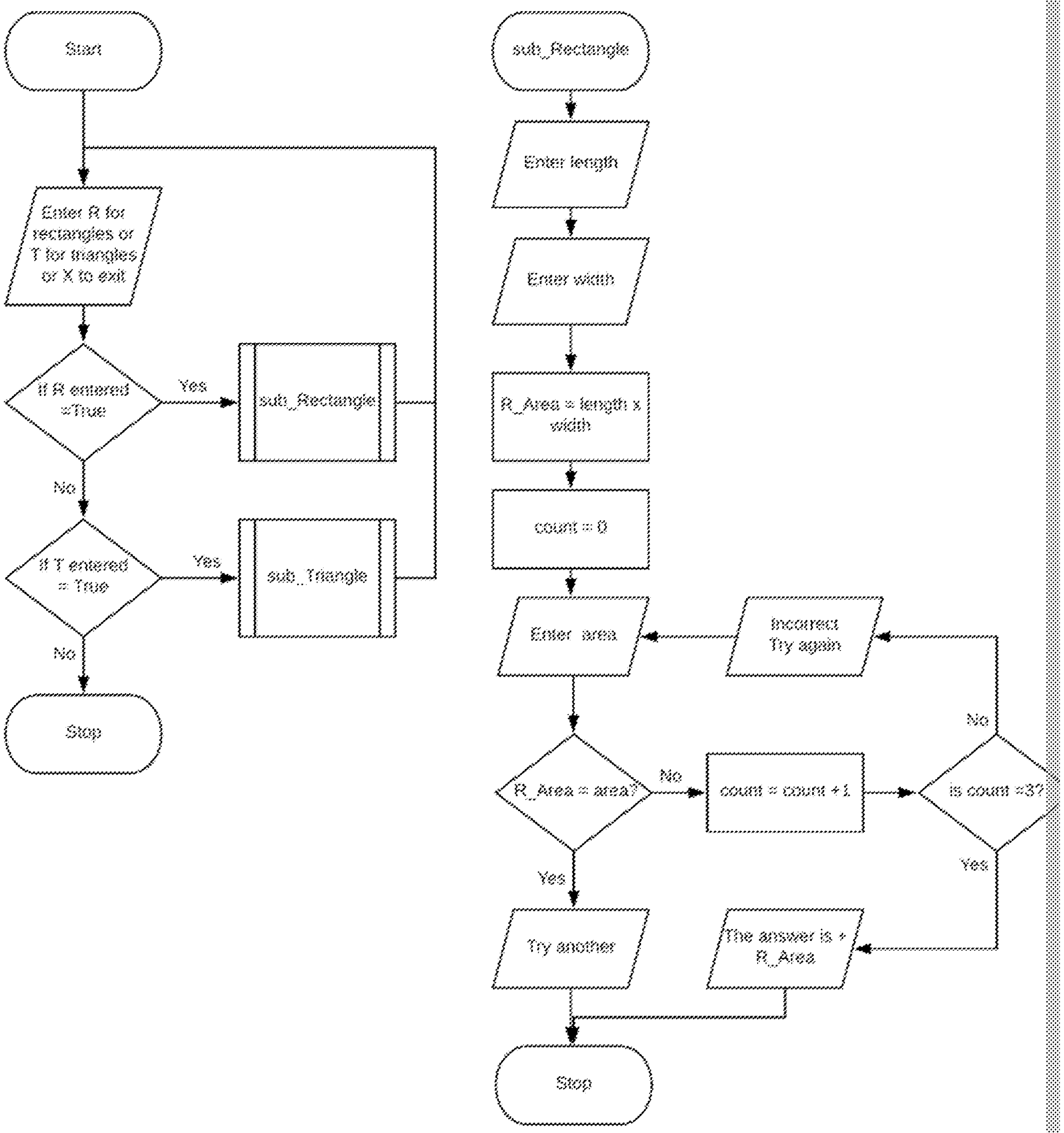
Note: The subprogram uses a PARAMETER in the design and uses an ARGUMENT (the ac subprogram is called.

```
function getString()
    validStr = false
    theString = input("Enter string")
    while NOT validStr
        if theString.length >= 10 AND theString.leng
            validStr = true
        else
            print("Incorrect- must be between 10 & 1
            theString = input("Enter string")
        endif
    endwhile

    return theString
endfunction

procedure getSubString(s)
    validStart = false
    validCharLength = false

    start = input("Enter start position")
    while NOT validStart
        if start < s.length AND start >= 0 then
            validStart = true
        else
            print("Not a valid number")
            start = input("Enter start position")
        endif
    endwhile

    charLength = input("Enter number of characters")
    while NOT validCharLength
        if start + charLength <= s.length then
            validCharLength = true
        else
            print("Not a valid number of characters"
            charLength = input("Enter number of char
        endif
    endwhile
    subStr = s.substring(start, charLength)

    print("Original string = " + s)
    print("Substring = " + subStr)
endprocedure

theString = getString()
getSubString(theString)
```

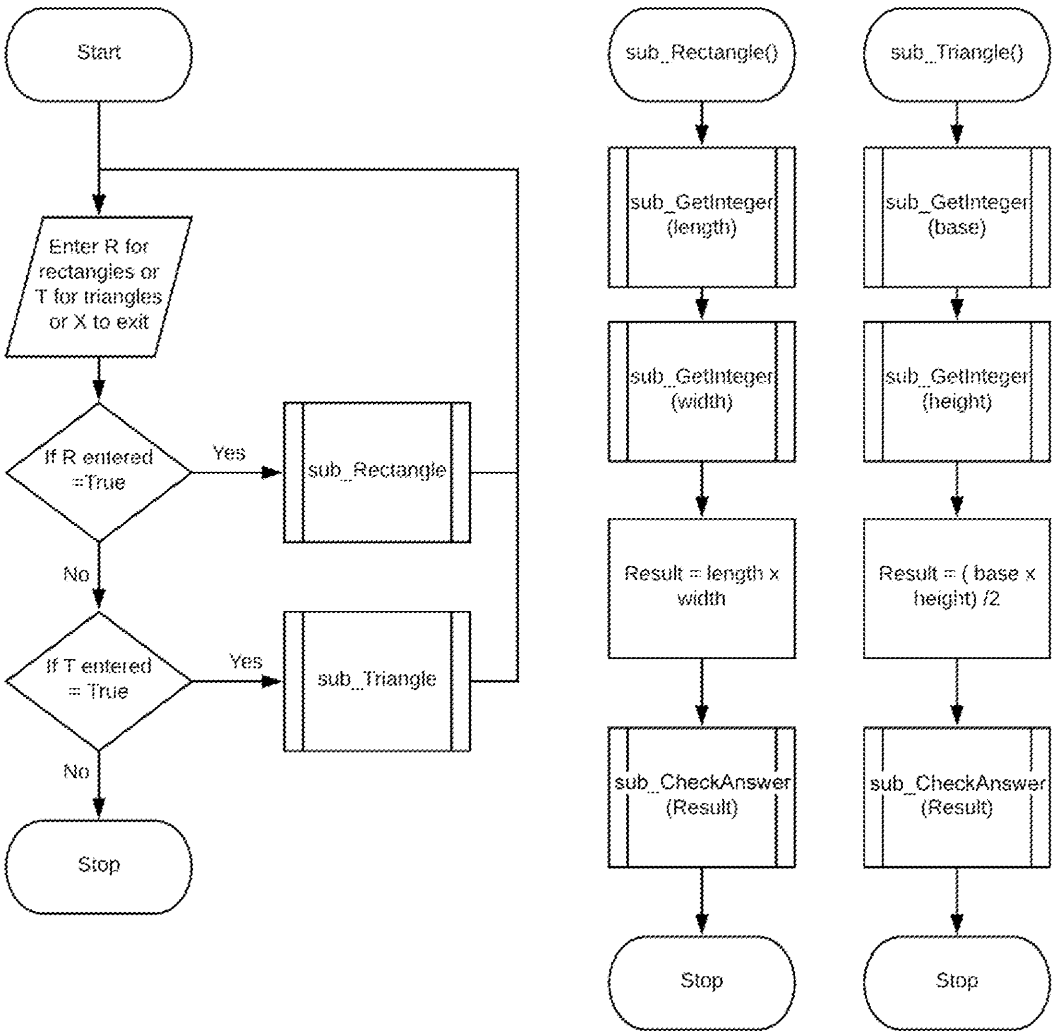
This is a para
a placeh

This is
ACTUAL v
sub

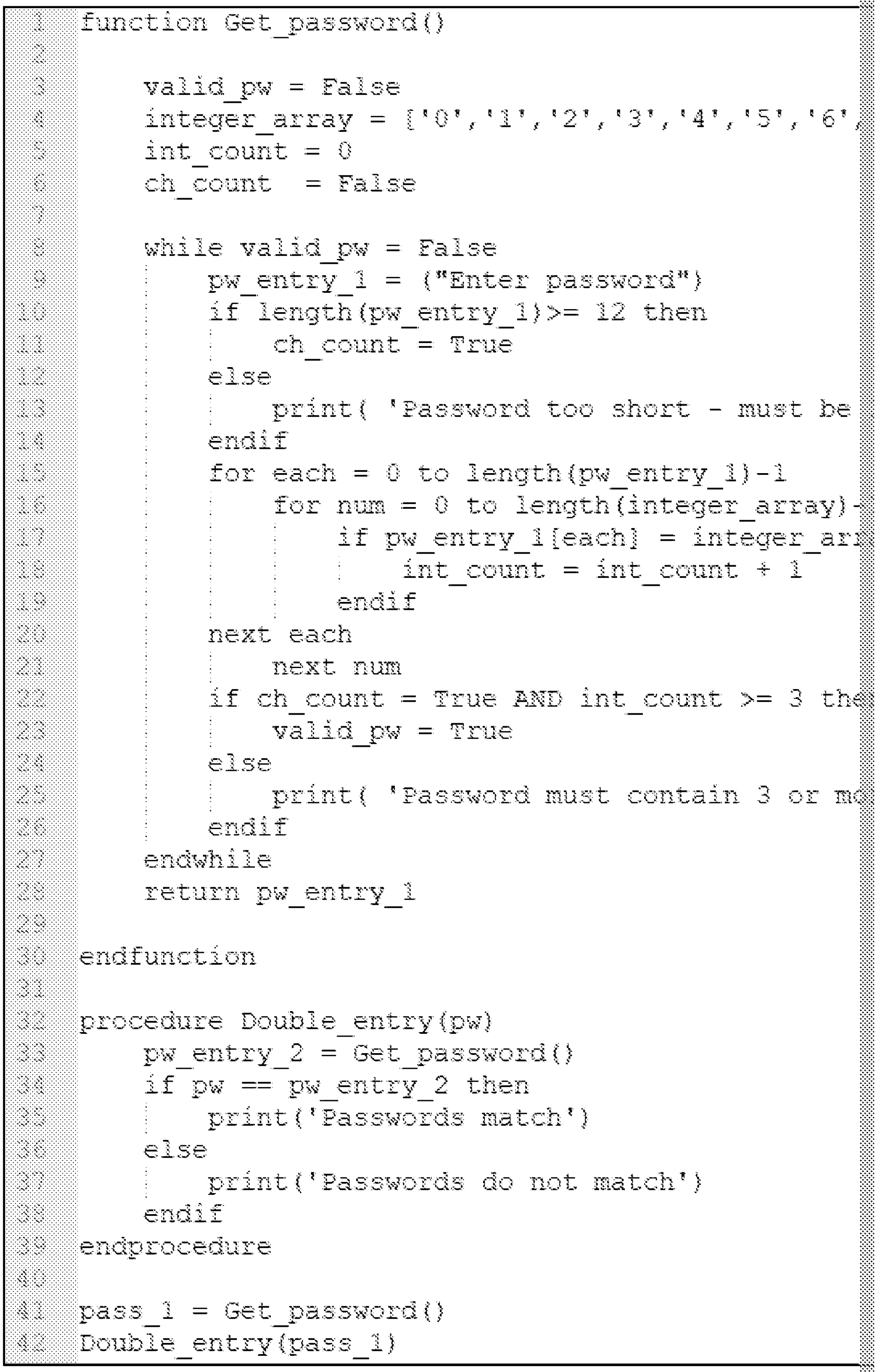
Exercise 17


Start
Enter R for rectangles or T for triangles or X to exit
If R entered =True
Yes
sub_Rectangle
No
If T entered = True
Yes
sub_Triangle
No
Stop
sub_Rectangle
Enter length
Enter width
R_Area = length x width
count = 0
Enter area
Incorrect Try again
R_Area = area?
No
count = count + 1
is count =3?
No
Yes
Yes
Try another
The answer is + R_Area
Stop

Alternative Solution

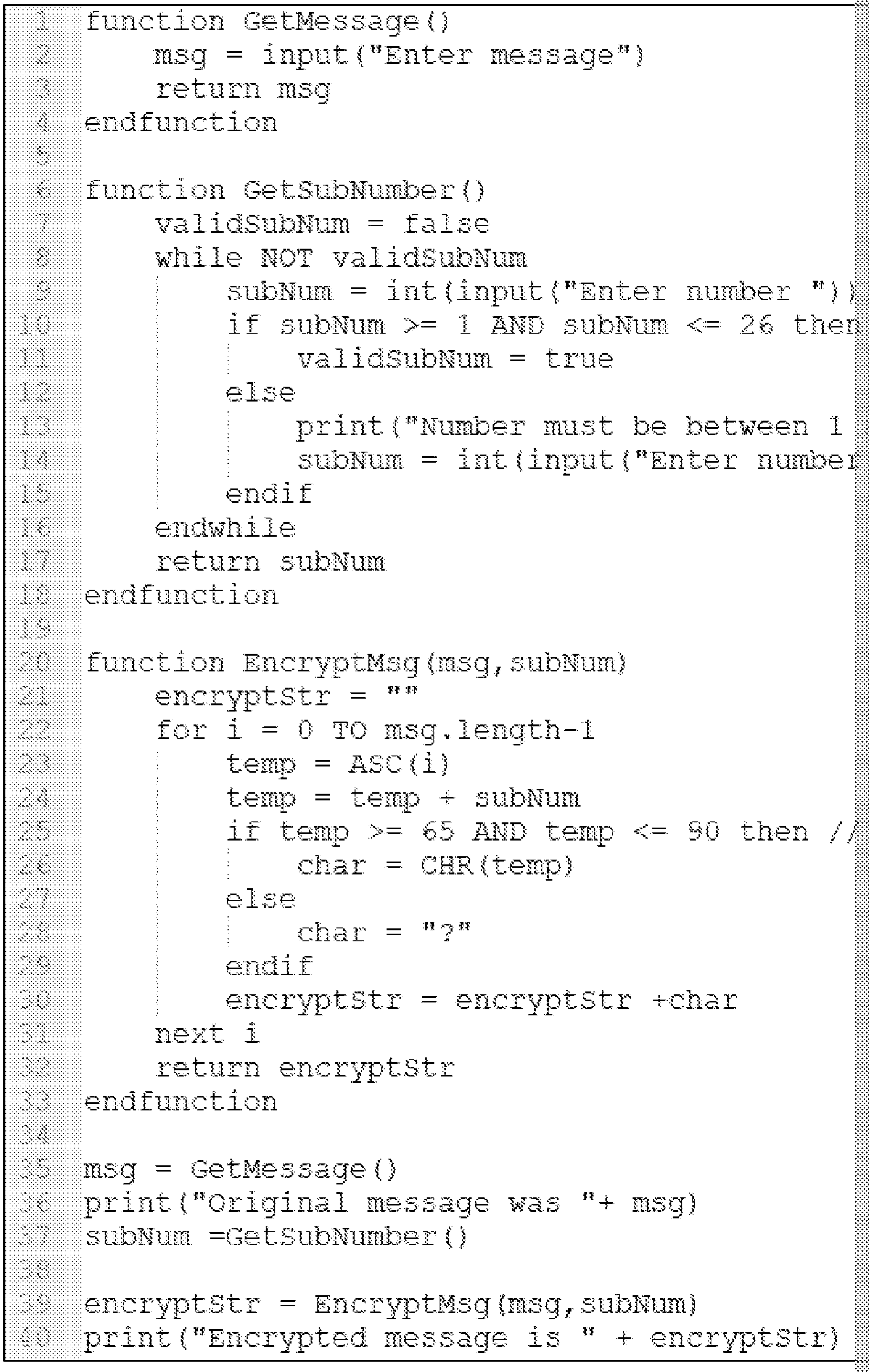


Exercise 18



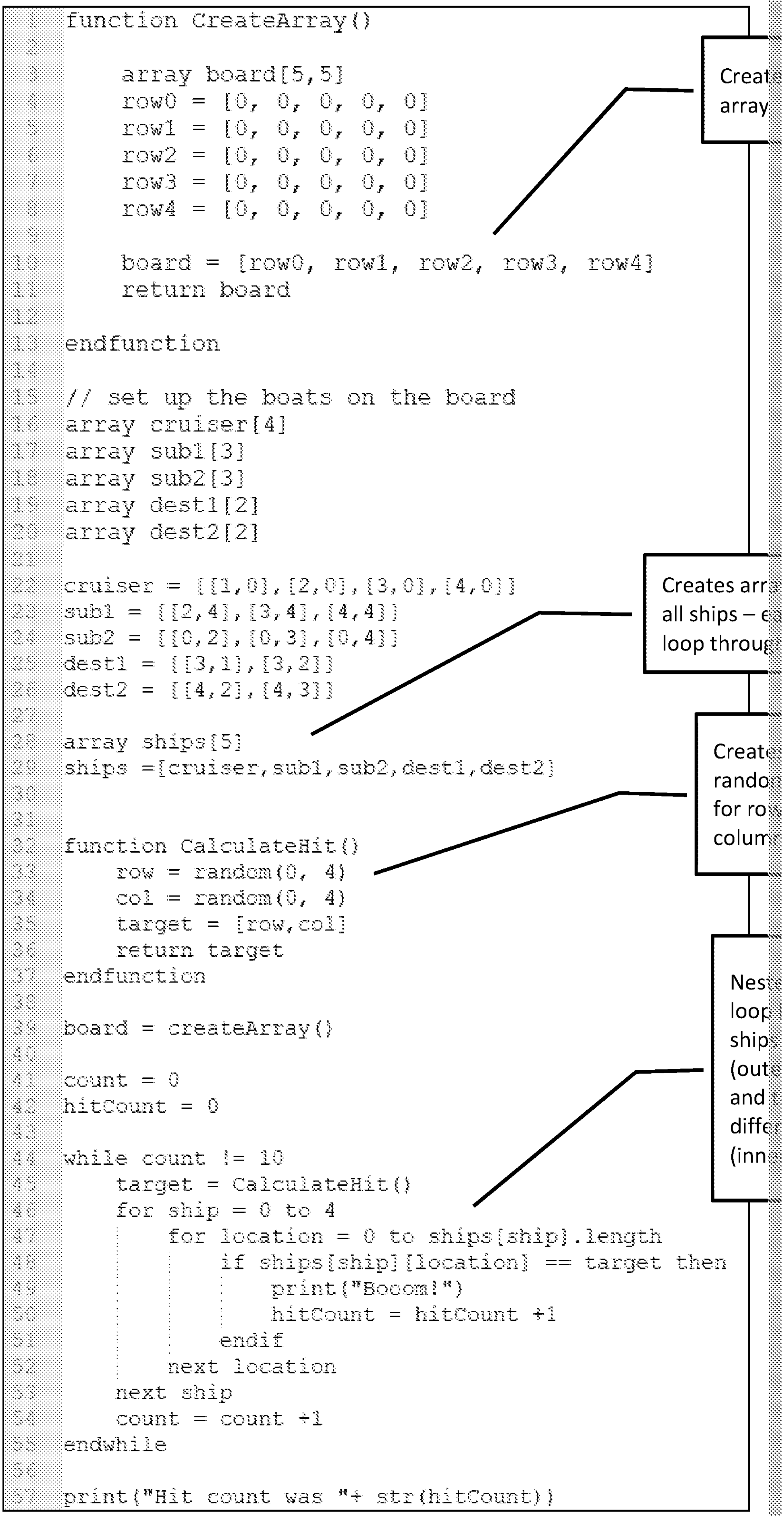
```
function Get_password()

    valid_pw = False
    integer_array = ['0','1','2','3','4','5','6',
    int_count = 0
    ch_count  = False

    while valid_pw = False
        pw_entry_1 = ("Enter password")
        if length(pw_entry_1)>= 12 then
            ch_count = True
        else
            print( 'Password too short - must be
        endif
        for each = 0 to length(pw_entry_1)-1
            for num = 0 to length(integer_array)-
                if pw_entry_1[each] = integer_arr
                    int_count = int_count + 1
                endif
        next each
            next num
        if ch_count = True AND int_count >= 3 the
            valid_pw = True
        else
            print( 'Password must contain 3 or mo
        endif
    endwhile
    return pw_entry_1

endfunction

procedure Double_entry(pw)
    pw_entry_2 = Get_password()
    if pw == pw_entry_2 then
        print('Passwords match')
    else
        print('Passwords do not match')
    endif
endprocedure

pass_1 = Get_password()
Double_entry(pass_1)
```

Exercise 19



```
function GetMessage()
    msg = input("Enter message")
    return msg
endfunction

function GetSubNumber()
    validSubNum = false
    while NOT validSubNum
        subNum = int(input("Enter number "))
        if subNum >= 1 AND subNum <= 26 then
            validSubNum = true
        else
            print("Number must be between 1
            subNum = int(input("Enter number
        endif
    endwhile
    return subNum
endfunction

function EncryptMsg(msg,subNum)
    encryptStr = ""
    for i = 0 TO msg.length-1
        temp = ASC(i)
        temp = temp + subNum
        if temp >= 65 AND temp <= 90 then //
            char = CHR(temp)
        else
            char = "?"
        endif
        encryptStr = encryptStr +char
    next i
    return encryptStr
endfunction

msg = GetMessage()
print("Original message was "+ msg)
subNum =GetSubNumber()

encryptStr = EncryptMsg(msg,subNum)
print("Encrypted message is " + encryptStr)
```

Exercise 20



```
function CreateArray()

    array board[5,5]
    row0 = [0, 0, 0, 0, 0]
    row1 = [0, 0, 0, 0, 0]
    row2 = [0, 0, 0, 0, 0]
    row3 = [0, 0, 0, 0, 0]
    row4 = [0, 0, 0, 0, 0]

    board = [row0, row1, row2, row3, row4]
    return board

endfunction

// set up the boats on the board
array cruiser[4]
array sub1[3]
array sub2[3]
array dest1[2]
array dest2[2]

cruiser = [[1,0],[2,0],[3,0],[4,0]]
sub1 = [[2,4],[3,4],[4,4]]
sub2 = [[0,2],[0,3],[0,4]]
dest1 = [[3,1],[3,2]]
dest2 = [[4,2],[4,3]]

array ships[5]
ships =[cruiser,sub1,sub2,dest1,dest2]


function CalculateHit()
    row = random(0, 4)
    col = random(0, 4)
    target = [row,col]
    return target
endfunction

board = createArray()

count = 0
hitCount = 0

while count != 10
    target = CalculateHit()
    for ship = 0 to 4
        for location = 0 to ships[ship].length
            if ships[ship][location] == target then
                print("Booom!")
                hitCount = hitCount +1
            endif
        next location
    next ship
    count = count +1
endwhile

print("Hit count was "+ str(hitCount))
```

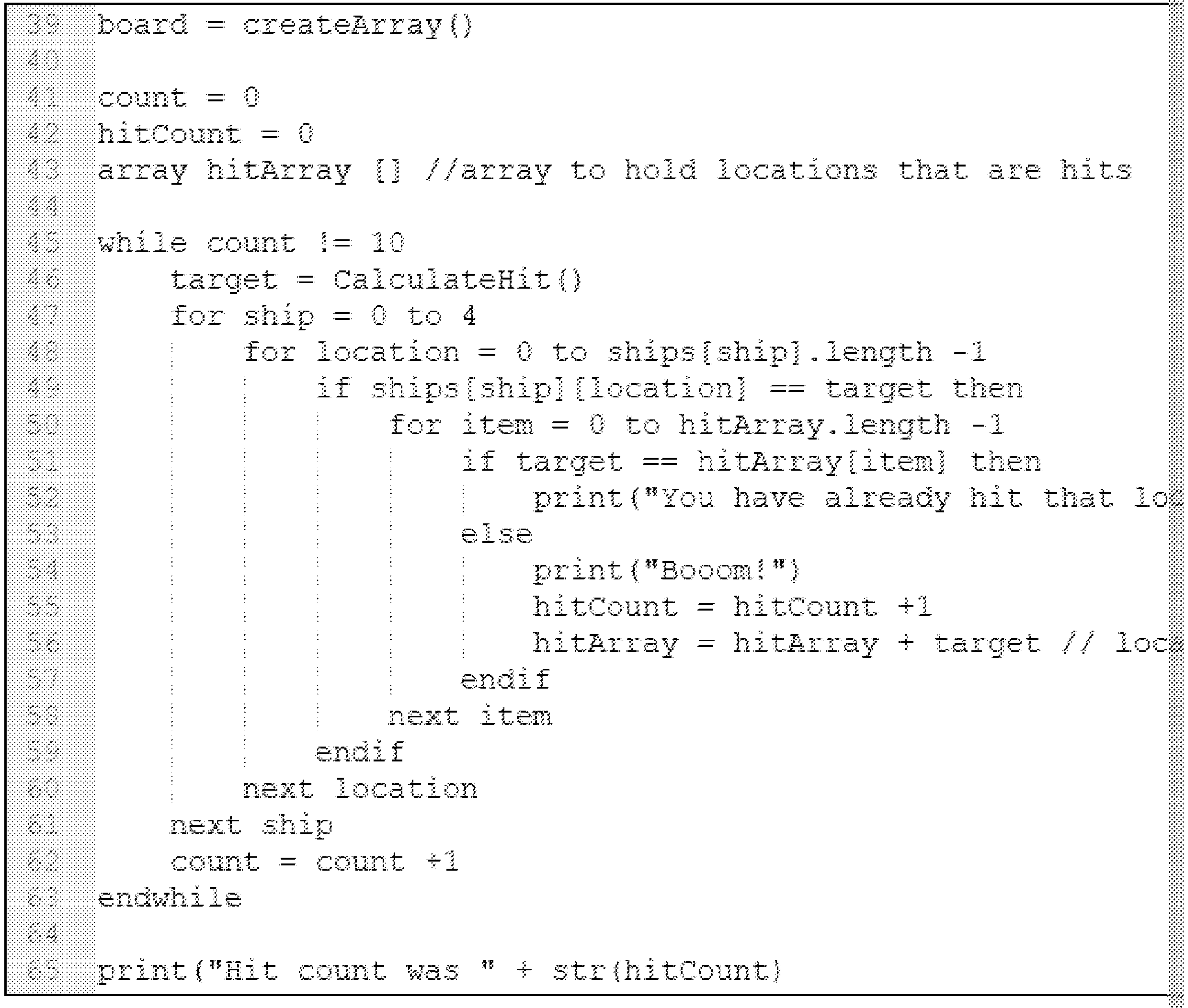
Creat
array

Creates arr
all ships – e
loop throug

Create
rando
for ro
colum

Nest
loop
ship
(out
and
diffe
(inn

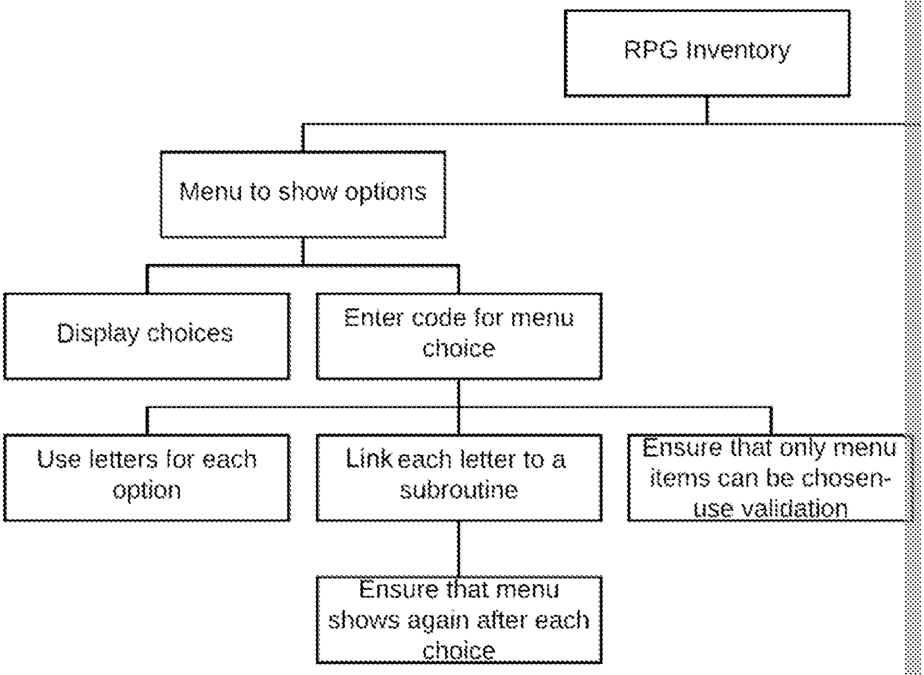
Exercise 20A



```
board = createArray()

count = 0
hitCount = 0
array hitArray [] //array to hold locations that are hits

while count != 10
    target = CalculateHit()
    for ship = 0 to 4
        for location = 0 to ships[ship].length -1
            if ships[ship][location] == target then
                for item = 0 to hitArray.length -1
                    if target == hitArray[item] then
                        print("You have already hit that lo
                    else
                        print("Booom!")
                        hitCount = hitCount +1
                        hitArray = hitArray + target // loc
                    endif
                next item
            endif
        next location
    next ship
    count = count +1
endwhile

print("Hit count was " + str(hitCount)
```

Exercise 21

Suggested plan for decomposing problem:


RPG Inventory
Menu to show options
Display choices
Enter code for menu choice
Use letters for each option
Link each letter to a subroutine
Ensure that only menu items can be chosen- use validation
Ensure that menu shows again after each choice

```
//runs the choices
procedure makeInventoryChoice(arr)
   menuOpt = DisplayMenu()

   while menuOpt != "X"
      if menuOpt == "D" then
           ViewInventory(arr)
           makeInventoryChoice(arr) // shows the me
      elseif menuOpt == "A" then
           arr = AddInventory(arr)
           makeInventoryChoice(arr) // shows the me
      elseif menuOpt == "U" then
           arr ← UseInventoryItem(arr)
           makeInventoryChoice(arr) # shows the men
      endif
   endwhile

   ExitInventory()

endprocedure

//display menu

function DisplayMenu()

   print("Enter D to view inventory")
   print("Enter A to add to inventory")
   print("Enter U to use an inventory item")
   print("Enter X to exit inventory menu")

   array menuChoice [4]
   menuChoice = ["D","A","U","X"]
   validChoice = False

```

```
        while NOT validChoice
            menuOpt = input("Enter choice ")
            for i ← 0 to menuChoice.length-1
                if menuOpt == menuChoice[i] then
                    validChoice = True
                elseif i == menuChoice.length-1
                    print("Please enter a valid menu o
                endif
            next i
        endwhile

        return menuOpt
    endfunction

    //View inventory

    procedure ViewInventory(arr)
        for i ← 0 to arr.length-1
            print(arr[i])
        next i
    endprocedure

    //Add item to inventory

    function AddInventory(arr)

        item = input("Name item to be added")
        arr = arr + item
        return arr

    endfunction

    //Use an inventory item

    function UseInventoryItem(arr)

        notFound = False
        item =input("What item do you want to use? ")
        for i = 0 to arr.length-1
            if item != arr[i] then
                notFound = True
                if notFound then
                    print("The item is not in the inve
                endif
            else
                print("You have now used this item")
                arr = arr-[item]
            endif
        next i
        return arr

    endfunction

    //Exit inventory menu

    procedure ExitInventory()
        print("You have exited the inventory menu")
    endprocedure

    // call subprograms to run inventory

    array inventoryArray  []
    makeInventoryChoice(inventoryArray)
```

Fox, chicken and grain problem

You must take the chicken across the river with you first.

A	B
FG	C

Next, take the fox across, leave it there and return with the chicken.

A	B
CG	F

Next, take the bag of grain across and leave it with the fox.

A	B
C	FG

Finally, return and take the chicken across.

A	B
	FCG

Exercise 22

You would think that the quickest way is to have Adam (1) carry the torch and do all the achieved by having Clair (5) and Danni (10) cross together.

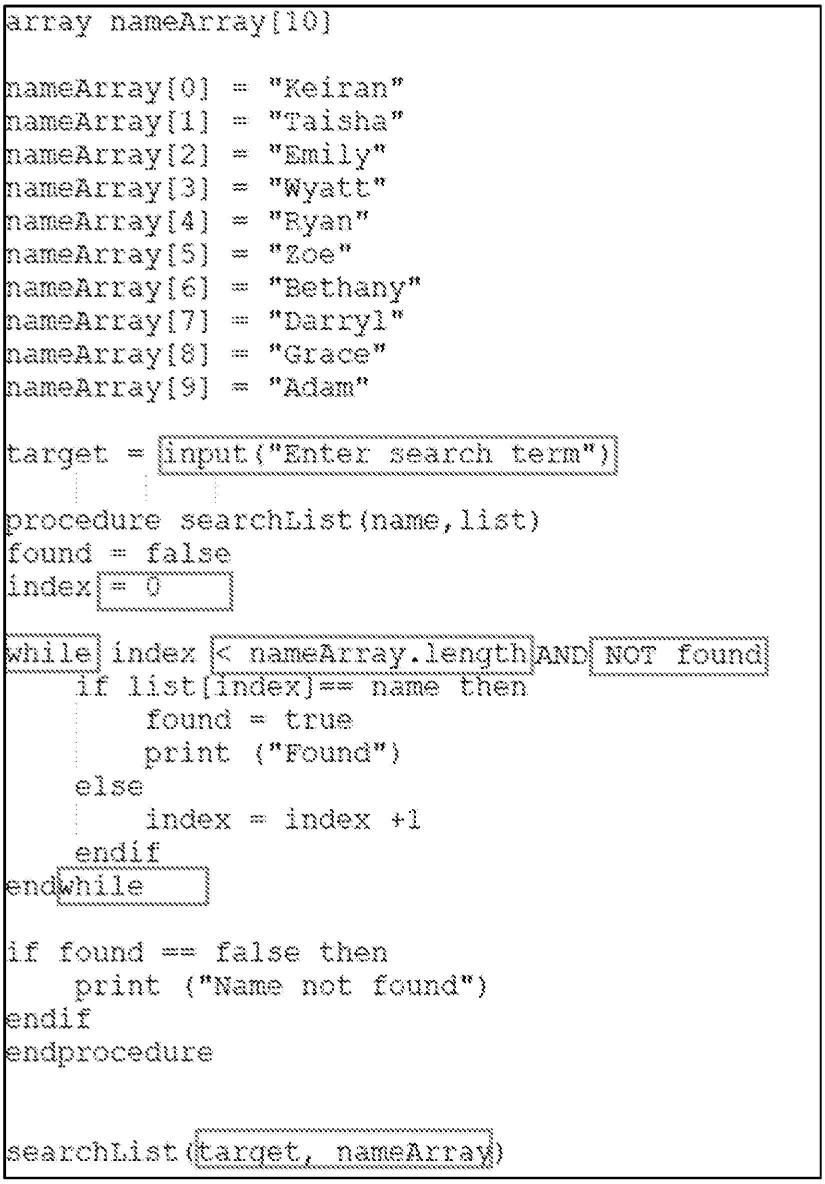
To simplify the solution, think about it like this first:

- A = 1
- B = 2
- C = 5
- D = 8

The moves are as follows:

Island	Bridge	Stage	Time Tal
C and D	A and B (with torch)	A and B	[illegible]
A, C and D	A returns (with torch)	B	[illegible]
A	C and D (with torch)	B, C and D	[illegible]
A, B	B returns (with torch)	C and D	[illegible]
	A and B (with torch)	A, B, C and D	[illegible]
		TOTAL	[illegible]

Exercise 23



```
array nameArray[10]

nameArray[0] = "Keiran"
nameArray[1] = "Taisha"
nameArray[2] = "Emily"
nameArray[3] = "Wyatt"
nameArray[4] = "Ryan"
nameArray[5] = "Zoe"
nameArray[6] = "Bethany"
nameArray[7] = "Darryl"
nameArray[8] = "Grace"
nameArray[9] = "Adam"

target = input("Enter search term")

procedure searchList(name,list)
found = false
index = 0

while index < nameArray.length AND NOT found
    if list[index]== name then
        found = true
        print ("Found")
    else
        index = index +1
    endif
endwhile

if found == false then
    print ("Name not found")
endif
endprocedure


searchList(target, nameArray)
```

Exercise 24

Linear search 1:

index	found	target	output
0	*False*	*34*	
1			
2			
3	True		Found at 3
4			
5			
6			
7			
8			
9			

Linear search 2:

index	found	target	
0	*False*	*1*	
1			
2			
3	True		F

Which is most efficient, and why?
Linear search 1 is less efficient as the FOR loop

```
for index = 0 to numsList.length
```

continues to loop through the array even when the search item has been found.

Linear search 2 uses a WHILE loop to check two conditions: whether the end of the array
search item remains not found. The WHILE loop will only continue whilst BOTH conditi
the search stops as soon as the item has been found.

Exercise 25

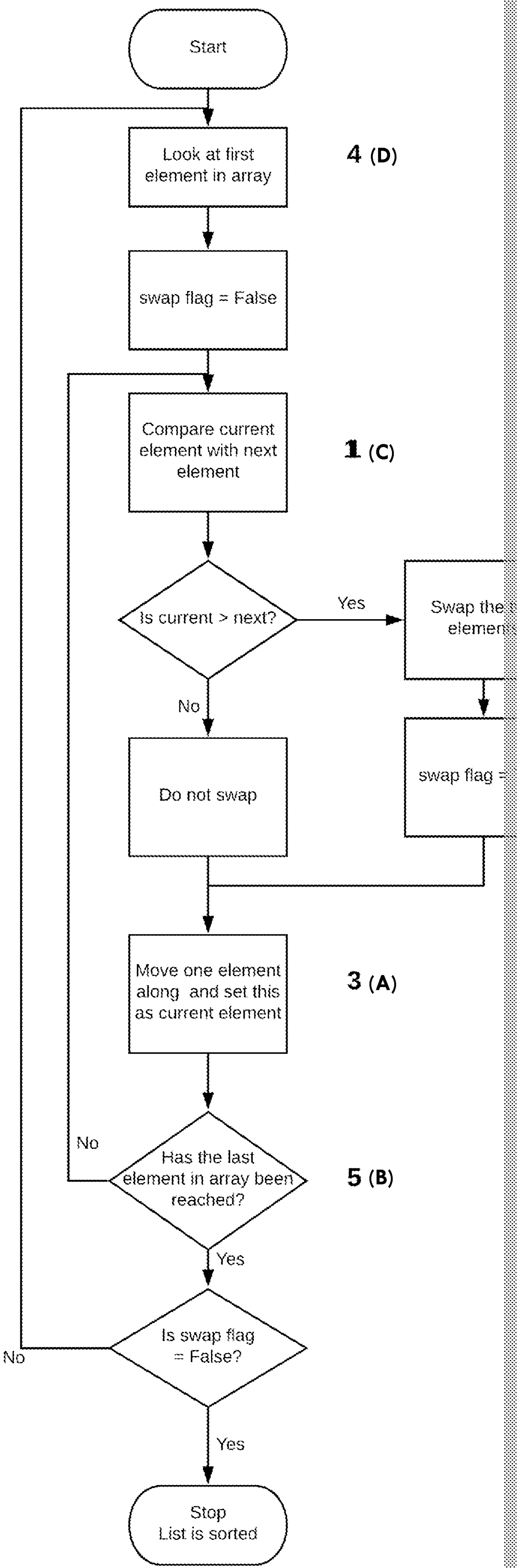
1.

5	1	6	2	4	3
1	5	6	2	4	3
1	5	2	6	4	3
1	5	2	4	6	3
1	5	5	4	3	6
1	2	5	4	6	3
1	2	4	5	3	6
1	2	4	3	5	6
1	2	3	4	5	6
1	2	3	4	5	6

Note: The final pass must be completed to confirm that no more swaps are needed a

2.
1. *Compare the first two elements in the array.*
2. *Is the first element bigger than the second element?*
3. If the answer is yes, the elements are swapped.
4. Move forward by one element and compare the current element with the one
5. Repeat steps 2, 3 and 4 until the end of the array is reached.
6. Repeat steps 1 to 6 until no swaps have been made.

Exercise 26


Start
Look at first element in array
4 (D)
swap flag = False
Compare current element with next element
1 (C)
Is current > next?
Yes
Swap the elemen
swap flag =
No
Do not swap
Move one element along and set this as current element
3 (A)
No
Has the last element in array been reached?
5 (B)
Yes
No
Is swap flag = False?
Yes
Stop
List is sorted

Exercise 27

1. (a)

```
                    67  23  52  6   15  43  11

            67  23  52  6           15  43  11

        67  23      52  6       15  43      11

        67      23      52      6       15

            23  67      6   52      15  43

            6   23  52  67      3   11  15  43

                    3   6   11  15  23  43  52
```

(b)

```
                    92  24  2   28  1   7   13

            92  24  2   28              1   7

        92  24      2   28              1   7

    92      24      2       28      1       7

        24  92      2   28              1   7

            2   24  28  92              1   7

                    1   2   7   12  13  24  28
```

2. a) Linear search, as the array is unordered.
   b) The algorithm takes in an array of data and a search term 'n'. The algorithm th sequentially to see if it matches the search term. When the whole array has be the variable **found** as True if the search term is in the array, or False if it is not
   c) Line 6 could be edited to incorporate the Boolean logical AND as follows: **WHI False**. This will make the algorithm more efficient as the WHILE loop will finish found.

3.
- Compare items [0] and [1] to see which is larger.
- Swap items so item [0] is smaller than item [1].
- Continue the comparison between item [1] and item [2].
- Swap the items so that item [1] is smaller than item [2].
- Repeat the process until the end of the array.
- Return to the start of the array and repeat again until no swaps are made.

4. The merge sort is a 'divide and conquer' algorithm which divides an unsorted array only contains one value before sorting and merging each pair, set of four, etc. It is a for very large data sets as it uses this division method. However, it requires exactly locations as the size of the data to perform the sort so it is very inefficient in terms

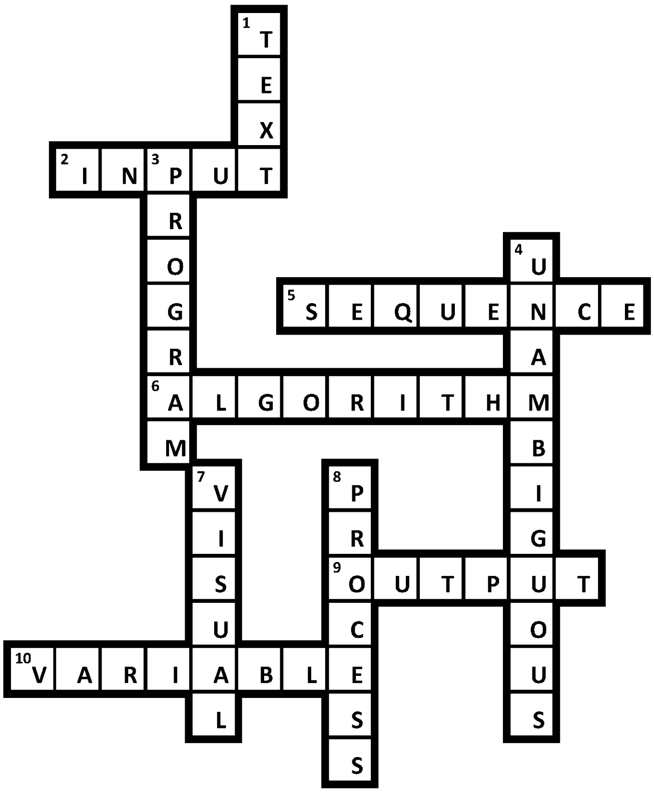
   The bubble sort is very efficient in its use of memory, only requiring one memory lo being swapped. Unlike the merge sort, the bubble sort works by comparing each pa the comparisons and swaps increases rapidly as the size of the data increases, maki time it takes.

5. The algorithm takes in an array of data as its parameter, starting at the first item in item. The algorithm then compares each item in the array with this initial value to c value of variable **smallest** is changed to the smaller value. When it has compared al smallest value.

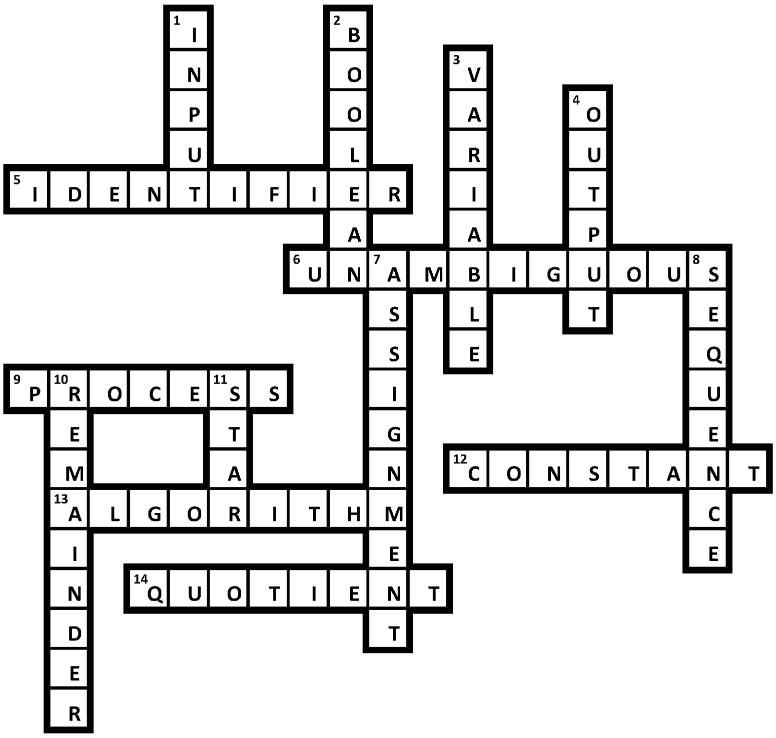
6. A binary search would be most suitable since the array is already sorted. A linear se very small and the time difference would be very small.

Crosswords

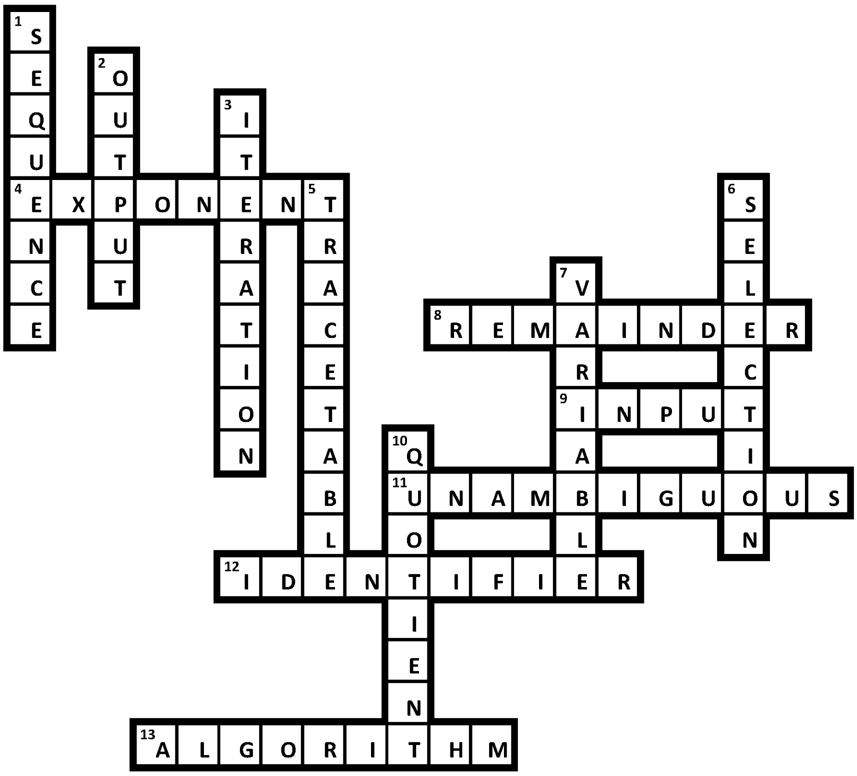
Crossword 1



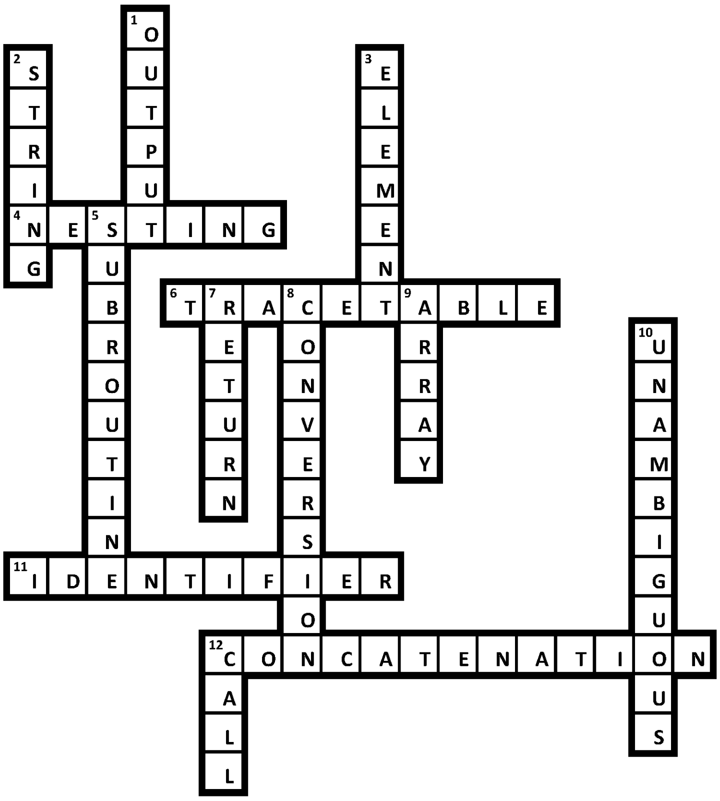
Crossword 2



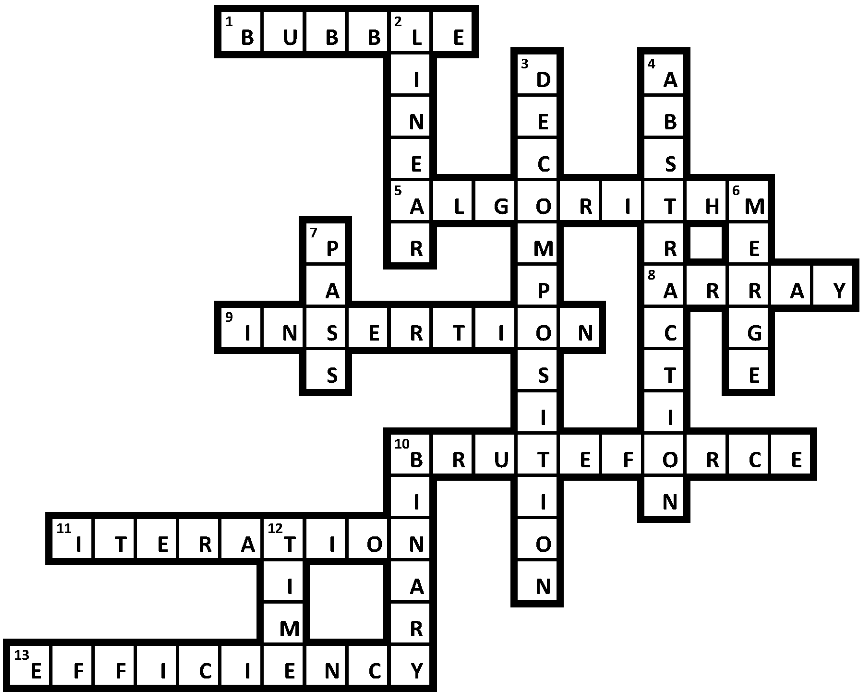
Crossword 3



Crossword 4



Crossword 5


1 BUBBLE
2 LINEAR
3 DECOMPOSITION
4 ABSTRACTION
5 ALGORITHM
6 MERGE
7 PASS
8 ARRAY
9 INSERTION
10 BRUTEFORCE
10 BINARY
11 ITERATION
12 TIME
13 EFFICIENCY