

STUDENT	
TEACHER	
CLASS	

WORKING AT GRADE	
TERM TARGET	
YEAR TARGET	



Year 9

Computing

2. Programming with Python

Pseudocode	Python	Description
BEGIN		Identifies the start of a program
END		Identifies the end of a program
READ X, Y, Z	input()	Identifies data that will be entered by the user
WRITE "X", Y, Z	print()	Identifies data that will be output by the program - note string values should be entered in speech marks
var = value	var = value	Assigns a value to a variable name
IF	if	Identifies the beginning of a section of code that will only be run if a condition is met
ELSEIF	elif	Identifies an alternative section of code that will only be run if a different condition is met
ELSE	else	Identifies the section of code that will run if no conditions are met
ENDIF		Identifies the end of the section of code that will run based on conditions
REPEATUNTIL	while True:	Identifies the beginning of a section of code that will continually execute until a condition is met
ENDUNTIL	break	Identifies where the until loop will terminate
PROCEDURE pname (X, Y, Z)	def pname (X, Y, Z)	Identifies the beginning and assigns a name to a subroutine (set of code) that will only run when the procedure is called
ENDPROCEDURE		Identifies the end of the code contained in the procedure
FUNCTION fname (X, Y, Z)	def fname (X, Y, Z)	Identifies the beginning and assigns a name to a subroutine (set of code) that will only run and return a value when the function is called
RETURN X	return X	Returning a value from within a function: computes the value of variable exits the function and returns the value of the variable
ENDFUNCTION		Identifies the end of the code contained in the function

NOTE: Your pseudocode should be indented in the same way as you would normally write a Python program i.e. inside if statements and loops. If you use BEGIN and END statements your whole program should be indented within the statements.

Progress against termly target												
ABOVE												
ON												
BELOW												
TERM	1		2		3		4		5		6	

Learning Outcomes			
	Levels		
Lesson	5	6	7
1	I can use a range of operators and expressions e.g. Boolean, and applies them in the context of program control.	I know and I can use negation with operators.	I know the effect of the scope of a variable e.g. a local variable can't be accessed from outside its function.
2	I have practical experience of a high-level textual language,	I know the need for, and can write, custom functions	I know and apply parameter passing.
3	I can represent solutions using structured notation.	I know the difference between, and I can use appropriately, procedures and functions.	I know and apply parameter passing.
4	I know that programming bridges the gap between algorithmic solutions and computers.	I can use nested selection statements.	I can apply a modular approach to error detection and correction
5	I know that iteration is the repetition of a process such as a loop.	I can find and correct syntactical errors.	I know the difference between, and I can use, both pre-tested e.g. 'while', and post-tested e.g. 'until' loops.
6	I can select the appropriate data types.	I can use and manipulate one dimensional data structures.	I can apply a modular approach to error detection and correction
7	Achieves a level 5 in the assessment	Achieves a level 6 in the assessment	Achieves a level 7 in the assessment

1. Python recap and negation



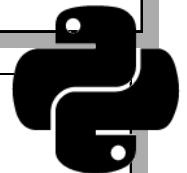
Complete the Sentence - programming keywords

Use the words in the list below to complete the sentence

1. An algorithm is a set of step by step _____ that solves a _____
2. Repetition means _____ an instruction a number of times
3. Debugging means to find _____ and _____ them
4. An _____ is data that is given to a _____ system
5. A _____ is a list of instructions that tell a computer what to do
6. Selection means that you have _____ choices to make in the program
7. A sequence is instructions that are placed in _____ and happen _____ after another
8. Storage is something that stores _____ for the computer to remember

program
repeating
different
order
errors
one
instructions
data
problem
fix
computer
input

Where you see the Python logo in this booklet you should copy the Python code into Python's IDE—IDLE (line numbers have been included to help you keep track of the code but should NOT be included when you copy the Python code into IDLE).



HINT: You must use the same indentation when copying Python code

1. `day = input("Please enter the day of the week:")`
2. `day = day.lower()`
3. `if day == "saturday" or day == "sunday":`
4. `print("Yay no school!")`



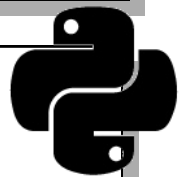
What does line 2 do?

HINT: turn the line of code into a comment by putting a # in front of it and try typing in the day of the week using capital letters.

What does line 3 do?

What programming construct from your do it now is this an example of?

HINT: This code should be added to the previous program as the line numbers continue from where the previous program finished



```
5. elif day == "wednesday":  
6.     print("Go to school but get home early!")  
7. else:  
8.     print("Go to school!")
```

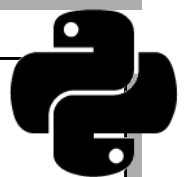


What does line 5 do?

When would line 8 be printed?

What is the difference between an if statement and an if, elif, else statement?

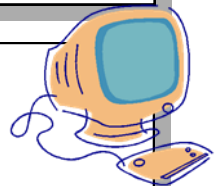
```
1. while True:  
2.     answer = input("At what temperature in degrees Celsius does water boil?")  
3.     if answer == "100":  
4.         break
```





This is an example of iteration in a program using a loop. How is iteration used in this program?

This kind of loop is called a post tested loop, using the example program explain why this is a post tested loop.



Read the following pseudocode.

```
BEGIN
  READ age
  IF age = 12 OR age = 13
    WRITE "You should be in year 8"
  ENDIF
END
```

Now write the program using Python, uploading it to Edmodo when you have finished.



Negation can be used to test if something is not equal to a condition.

When using relational operators (equal to, more than, less than) you use an exclamation mark to write the operator in its negated form i.e. `!=` means not equal to.

How would you write 'not more than'?

You can also use negation with Boolean operators (and, or, not). To do this all you do is put the conditions that you want to negate into brackets and put a not in front of them i.e.

`not (name == "Jane" or name == "Anne")` would test that name is not equal to 'Jane' or 'Anne'.

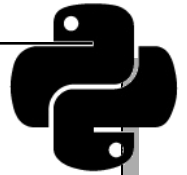
How would you write the line of code to test if a username and password had been entered incorrectly

How do you write not equal to?

When you have initialised variables in the examples so far you have only used them within the main method of the Python program. You are going to be starting to write your own functions and procedures so you will need to be able to use your variables wherever they are needed in the program. To do this you need to understand local and global variables.

Global variables can be accessed anywhere in the program i.e. they can be used globally.

Local variables can only be accessed inside the function or procedure they are written in.



```
1. def simonSays():
2.     simon = " "
3.     global simon
4.     peter = " "
5.
6.     simon = input("What does simon say")
7.     peter = input("What does peter say")
8.
9. simonSays()
10. print("Simon says",simon)
11. print("Peter says",peter)
```

What does line 3 do?

Which variable is global and which is local?

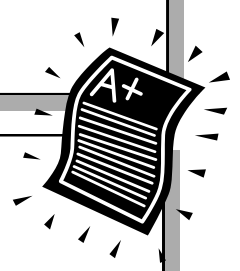
What happens when you run the program?

Why do you think this happens?

Adapt the program so that both print statements are executed. Can you think of a way of adapting the code so that both are printed without making peter a global variable? Upload your new version of the program to Edmodo.



Self Assessment: R A G	Exit Ticket: What do we mean by negation?
--------------------------------------	---



2. Sub-solution using procedures

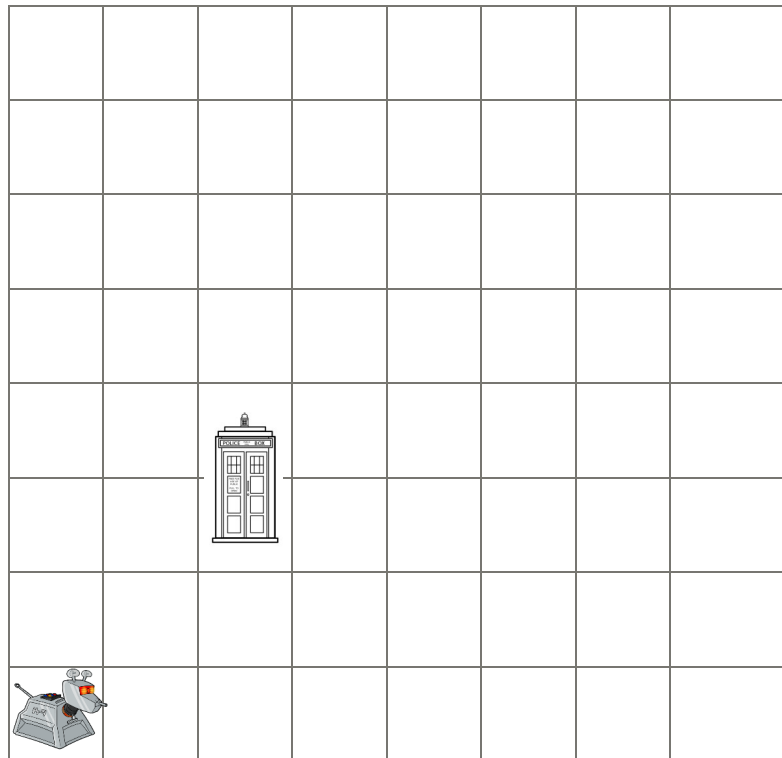


K9 is Dr Who's dog and he has got separated from the TARDIS—can you help him find his way back?

K9 can only use the following commands:

Command	Description
forward (number of squares)	Move forward a set number of squares
right()	Turn right 90°

Write instructions for K9 to find the TARDIS.



To make programming K9 easier The Doctor has created a way to teach K9 new commands. To do this you need to write the name of the new command followed by brackets and a colon and then all of the instructions needed to complete the new command. Now you can use the new command each time you need it in the program.

For example:

2step():

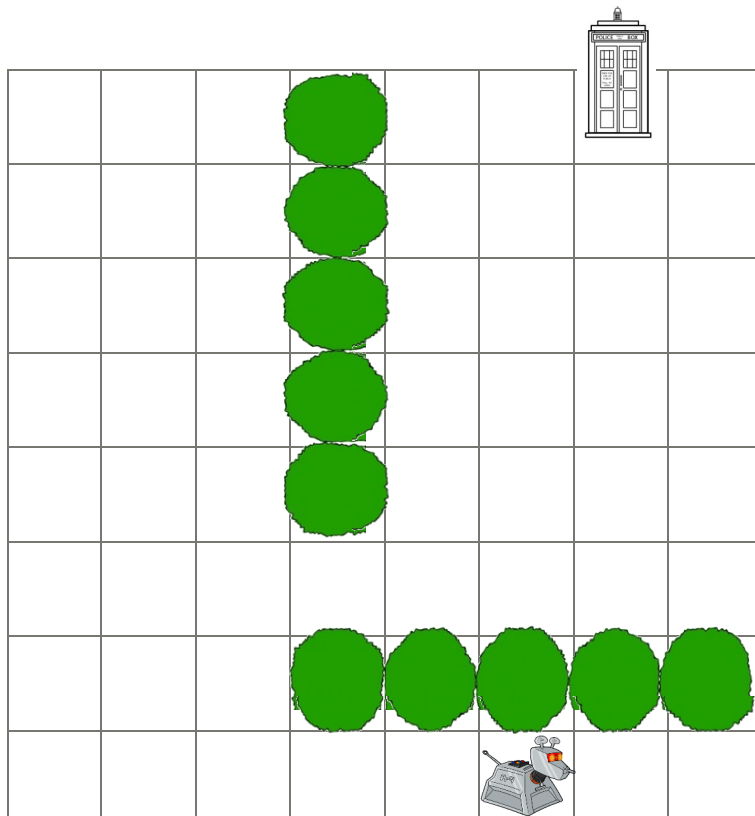
forward(1)

forward(1)

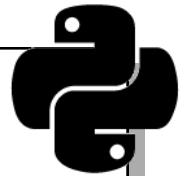
right()

2step()

Now write a new program for K9 to find the TARDIS:



What you just created for K9 is called a procedure and these can be used to hide detail in programs. This is called sub-solution. When a value is returned this is called a function. Below is an example of a program that uses a function to calculate VAT.

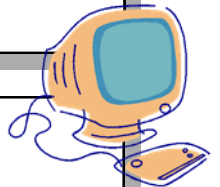


```
1. def VAT():
2.     priceIncVAT = (price / 100 )*120
3.     return priceIncVAT
4. price = input("Please enter the price of the item:")
5. price = float(price)
6. print(VAT())
```

What does the function in the example calculate?



The program is going to be extended so that 10 item prices can be input and each item's total price output on the screen. Explain why the use of a function makes extending the program simpler.



A business that sells carpet uses the following formula to work out the price for customers:

carpetPrice = (width of room X depth of room) X price per square metre

NOTE: in Python * is used for the multiplication symbol

Write a function that the business could use to calculate the price of customers carpet orders and then use it in a program for making quotes for customers. Upload onto Edmodo.

A function always does what:



```
1. def VAT(price):
2.     priceIncVAT = (price/ 100) * 120
3.     return priceIncVAT
```



This function also works out a price including VAT, however this example also uses parameters (in this example price). Parameters are values passed into a function or parameter.



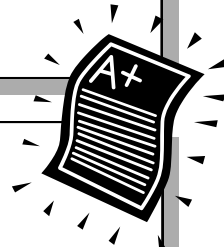
4. `price = input("Please enter the price of the item:")`
5. `price = float(price)`
6. `print(VAT(price))`

As you can see from the example when you called the function you also had to pass in the parameters—which when you call a function are called arguments. You will also notice that you have to tell Python what you want to do with the return value—in this case it has been printed. **NOTE:** Whenever you call the function in the program you will execute the return value but this will only be output on screen if the function is called inside a print function.

You can estimate how many minutes someone has lived using the following calculation:

```
ageInMins = ((age * 365) * 24) * 60
```

Create a program using a function that calculates a users age in minutes and outputs the value on screen. Upload your program onto Edmodo.



Self Assessment: R A G	Exit Ticket: What is a function used for?
----------------------------------	---

3. Writing Psuedocode



I have written the following program using Python but it doesn't seem to be working correctly. There are 8 syntax errors (errors that mean my code won't work) in my code can you circle them all? **HINT:** Each line has one error in it.

1. `fed BMI():`
2. `BMlcalcuation =(weightInPounds / (heightInInches * heightInInches)) x 703`
3. `return BMlcalcuatio`
4.
5. `weightInPound = input ("What is your weight in pounds?")`
6. `weightInPounds == int(weightInPounds)`
7. `heightInInches = input ("What is your height in inches?")`
8. `heightInInche = int(heightInInches)`
9.
10. `print(bmi())`

Pseudocode is used as a way of planning a computer program (or algorithm) - it is a form of structured notation which another person could follow to write the program using computer code.

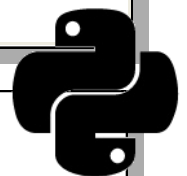
BEGIN

All keywords in your pseudocode should be written in **CAPITAL LETTERS** - best way to think about a keyword is that it is an **ACTION** word from your code i.e. **READ, IF, SUM** etc. You should also start your program with **BEGIN** and end your program with **END**. See table on opposite page.

Pseudocode	Python	Description
BEGIN		Identifies the start of a program
END		Identifies the end of a program
READ X, Y, Z	input()	Identifies data that will be entered by the user
WRITE "X", Y, Z	print()	Identifies data that will be output by the program - note string values should be entered in speech marks
var = value	var = value	Assigns a value to a variable name
IF	if	Identifies the beginning of a section of code that will only be run if a condition is met
ELSEIF	elif	Identifies an alternative section of code that will only be run if a different condition is met
ELSE	else	Identifies the section of code that will run if no conditions are met
ENDIF		Identifies the end of the section of code that will run based on conditions
REPEATUNTIL	while True:	Identifies the beginning of a section of code that will continually execute until a condition is met
ENDUNTIL	break	Identifies where the until loop will terminate
PROCEDURE pname (X, Y, Z)	def pname (X, Y, Z)	Identifies the beginning and assigns a name to a subroutine (set of code) that will only run when the procedure is called
ENDPROCEDURE		Identifies the end of the code contained in the procedure
FUNCTION fname (X, Y, Z)	def fname (X, Y, Z)	Identifies the beginning and assigns a name to a subroutine (set of code) that will only run and return a value when the function is called
RETURN X	return X	Returning a value from within a function: computes the value of variable exits the function and returns the value of the variable
ENDFUNCTION		Identifies the end of the code contained in the function

NOTE: Your pseudocode should be indented in the same way as you would normally write a Python program i.e. inside if statements and loops. If you use BEGIN and END statements your whole program should be indented within the statements.

1. **def question():**
2. **while True:**
3. **answer = input("Who was Henry the 8th's first wife?")**
4. **if answer == "Catherine of Aragon":**
5. **break**
6. **question()**





The program on page 10 is an example of a procedure. Explain what the difference between a function and a procedure:

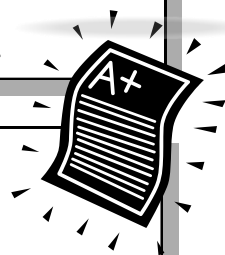
Using the Pseudocode table on page 10 write this program using Pseudocode:

Rewrite the pseudocode so that it uses a function with the return value "Your answer is correct"

Rewrite the program on page 10 so that you pass in parameters for the question and answer.

Upload your program onto Edmodo.

State whether you will use a function or procedure and why: _____



Self Assessment:

R A G

Exit Ticket: What does a procedure not have?

4. Selection Statements



The suspects :

Professor Plum

Miss Scarlet

Mr Black

Reverend Green

Mrs Peacock

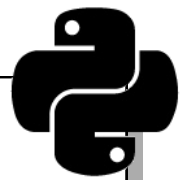
Doctor Pink has been found murdered downstairs in the Kitchen, battered by some lead piping, at 20:00 hours. By a process of elimination find out **whodunnit** by looking at the following pseudocode.

HINT: <> means not equal to

```
IF TimeFound < 21:00 THEN
    IF (Location = Downstairs) AND (weapon=knife) THEN
        Reverend Green = innocent
    END IF
    IF (Location = Downstairs) OR (weapon=knife) THEN
        Mr Black = innocent
    ELSE
        Miss Scarlet=Innocent
    ENDIF
ELSE
    Professor Plum = innocent
END IF

IF NOT (Location=Conservatory) THEN
    IF Location=Kitchen THEN
        CASE Weapon of
            Gun: Mr Black=innocent
            Candlestick: Reverend Green = innocent
            Lead Piping : Mrs Peacock = innocent
            Rope : Professor Plum=innocent
        END CASE
    END IF
    IF (location <> upstairs) AND (TimeFound>12:00) Then
        Professor Plum = innocent
        IF location=kitchen THEN
            Miss Scarlet=innocent
        ENDIF
    ELSE
        Mr Black = innocent
    ENDIF
END IF
```

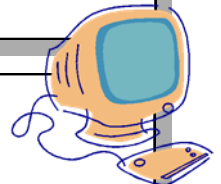
The murderer was _____



```

1. print("What flavour milkshake would you like:")
2. print("1. Banana")
3. print("2. Chocolate")
4. print("3. Strawberry")
5. while True:
6.     choice = input("Please enter a number")
7.     if choice == "1" or choice == "2" or choice == "3":
8.         break

```



Extend the above program using the following pseudocode for guidance:

BEGIN

```

IF choice = "strawberry"
    WRITE "Enjoy your strawberry milkshake"
ELSEIF choice = "chocolate"
    WRITE "Enjoy your chocolate milkshake"
ELSE
    WRITE "Enjoy your banana milkshake"
ENDIF

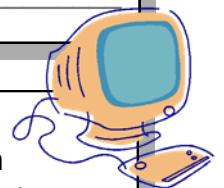
```

END

Upload your completed program to Edmodo.



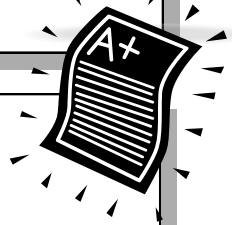
How does programming bridge the gap between algorithmic solutions and computers?



Selection statements can also be used inside each other (see the do it now task for an example), this is called **nesting**. Extend your milkshake program to include nested selection statements which ask the user if they want chocolate sprinkles if they chose a chocolate, fresh strawberry if they chose strawberry or banana pieces if they chose banana.



I have uploaded a program on Edmodo with some syntax errors - which procedure or function contains the syntax error?

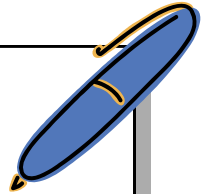


Self Assessment:
R A G

Exit Ticket: What is a selection statement?

STRENGTH	TARGET	ACTION	EFFORT

Green Pen Activity:



5. Iteration



BEGIN

```
countdown = 10
```

```
REPEATUNTIL countdown = 0
```

```
    countdown = countdown -1
```

```
    WRITE (countdown )
```

```
ENDUNTIL
```

```
WRITE "Blast off"
```

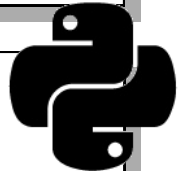
END

How many times will the UNTIL loop repeat? Why?

What number will be written first? Why?

What number will be written directly before "Blast Off"? Why?

1. `age = input("How old are you?")`
2. `age = int(age)`
3. `while True`
4. `age = input("You are too young. How old are you?")`
5. `age = int(age)`
6. `if Age >=18:`
7. `break`



Correct the syntax errors (3) in the code and upload to Edmodo.

This program uses relational operators to terminate the program. Which line stops the loop?

Explain under what condition the loop stops.

How would you change the program so it repeats until the user says they are under 18?



What is iteration?



1. `countdown = 11`
2. `while countdown >1:`
3. `countdown = countdown - 1`
4. `print(countdown)`
5. `print("Blast off")`

I have created the above program using a while loop. This is sometimes known as a pre-tested loop. Explain why this is a pre-tested loop:

What kind of loop is an until loop?

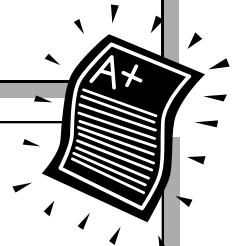
Change the program so that (still using a while loop) it outputs the 5 times table up to 50.



Self Assessment:

R A G

Exit Ticket: What is iteration?



6. Data types

Complete the Sentence

Use the words in the list below to complete the sentence

1. Iteration A group of _____ is executed repeatedly until a condition is met (a _____)
2. Selection The pathway through a program is selected by using a _____ to decide on what instructions to _____ next.

condition
instructions
execute
loop).



When you write code you need to use different data types for different purposes. When you convert one data type to another this is called **type casting**. When you use the input function this is defaulted to store string data types - if you want to perform any math on the data you must type cast it first.

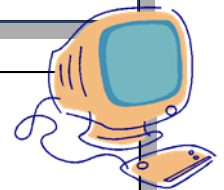


Fill in the table with the relevant python code and an example of the data for the type

Data type	Python code	Example data
Integer		
Real / float		
String		

Write a quiz program containing at least 5 questions. This must include at least one procedure and type casting where appropriate (I expect at least one example of each data type included in the answers required by the quiz).

Upload your completed quiz program to Edmodo.



So far we have only used basic data structures. We will now begin to look at more complex structures. The first is called an array (in Python this is called a list). To initialise an array in python you begin with the name you want to give the array followed by the items in the array:

```
arrayName = [ "data", "data", "data"]
```

You can use all three data types in an array but each item in the array must be of the same data type.

To initialise an empty array ready to add data to you just name the array and open and close square brackets:

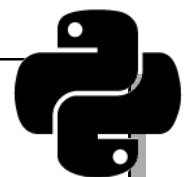
```
arrayName = [ ]
```

Now you can use the following code to add to your array:

```
arrayName.append(data to be appended)
```

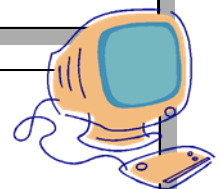
Each time you use this line of code a new item will be added to the end of the array.

```
1. shoes = [ ]
2. while True:
3.     shoeType = input("Please enter shoe type:")
4.     shoes.append(shoeType)
5.     if len(shoes) == 3:
6.         break
7.     print(shoes)
```



Using the above code to help you, extend your quiz program using an array to generate a score. The players score should be displayed on screen at the end of the game.

NOTE: `sum(arrayName)` will add up the data in the array, `len(arrayName)` will return the number of items in the array. Upload to Edmodo.



7. End of unit assessment



Programming Keywords

Can you find all of the programming keywords listed below.

V S T R I N G J L Y D A M J R Y R V F Z
 V X D T S M P H P R O C E D U R E F Y J
 Y W U E T M U F F G U Y D F A D G Y D R
 D C C L S Q H T E E B Y F C Z S E D B Y
 E Z H B E E Y T R C T D M T F D T L W D
 T Z W A Q Z N P I E O A E R P T N H M C
 T H B I U L O I Y R T T W C V Q I A O V
 M J Q R E G G P T U O C C C I X E N M I
 K K C A N O R U S U Z G A O P S S R G D
 J Z B V C I M F A E O L L R X T I V A S
 C N I C E V Z O T F E R B A A J B O M N
 L O M T O Y F E J S G O B N W H L J N S
 L I T N E M E T A T S N T U Q E C K J E
 M T N L X R L A J F V A O A S W L M I L
 B A E E R E A L K C Q J U I R F B L O E
 M R W E S M H T I R O G L A T R P C B C
 R E B Y K T B F I U R J P D K C A T C T
 K T C O L F I T F O B O O L E A N Y F I
 T I Y D G R X N A T N J I V X R Z U A O
 S S E C O R P X G Z N J H V B Z C N F N

ALGORITHM
 FUNCTION
 SEQUENCE
 ITERATION
 NESTING
 CHARACTER
 REAL

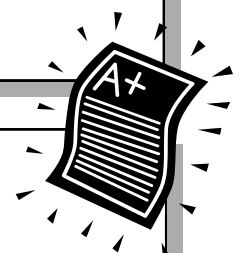
DECISION
 PROCEDURE
 STATEMENT
 ALGORITHM
 ARRAY
 CONSTANT
 STRING

PROCESS
 SELECTION
 SUBROUTINE
 ITERATION
 BOOLEAN
 INTEGER
 VARIABLE

Self Assessment:

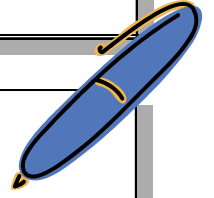
R A G

Exit Ticket: How do you think you did?



STRENGTH	TARGET	ACTION	EFFORT

Green Pen Activity:



Keywords



Pseudocode	A description of a computer programming algorithm that uses the structural conventions of a programming language, but is intended for human reading rather than machine reading.
Algorithm	A set of rules specifying a how to solve a problem.
Decision	A selection from a range of options depending upon the result of a condition.
Process	To perform logical operations on (data) according to programmed instructions in order to achieve a desired result.
Function	A subroutine that executes the statements and returns a single value to the program.
Procedure	A subroutine that executes the statements and returns control to the program.
Selection	The pathway through a program is selected by using a condition to decide on what instructions to execute next.
Sequence	Set of instructions to be carried out in the order they are written.
Statement	A single instruction or step within a program.
Subroutine	A subset of code within a larger program, which performs a specific task.
Assignment	Sets or resets the value stored in the storage location denoted by a variable name.
Algorithm	A set of rules specifying a how to solve a problem.
Iteration	A group of instructions is executed repeatedly until a condition is met (a loop).
Nesting	When control structures are inserted within other control structures.
Array	A block of variables of the same type using a single name and an index value.
Boolean	Variables that store just two values, e.g. TRUE or FALSE.
Character	Data type that stores a single character.
Constant	Name used to identify a value in memory that does not change during the execution of the program.
Integer	Whole number values, positive or negative.
Real	Data type that will store decimal (or fractional) values.
String	Data type used to store a string of characters.
Variable	Name used to identify a value in memory that can change during the execution of the program.