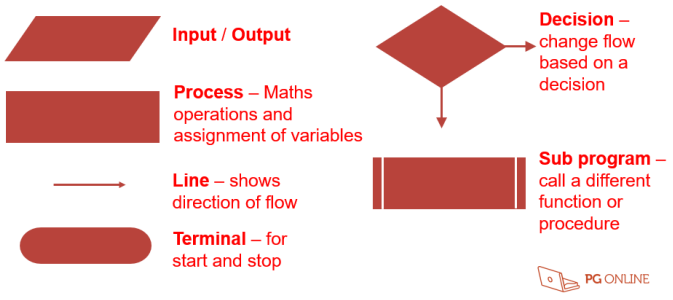
Year 10 Keywords: Spring 2

Topic Title: ALGORITHMS	
Keyword	**Definition**
Abstraction	Removing unimportant parts of a problem in order to concentrate on those that are important
Decomposition	Breaking down a problem into smaller more manageable ones
Algorithmic thinking	An approach to solving problems by the use of algorithms (sequences of steps that lead to a solution)
Structure diagram	A hierarchical diagram that shows how a problem is broken down into sub-sections/sub-tasks
Binary search	This only works on a sorted list The middle item of the list is first checked If the item searched for is less than this item the right of the list is discarded, and a binary search is carried out on the left of the list
Linear search	Each item in the list is checked against the search item in order
Sorting algorithms	• Bubble sort • Insertion sort • Merge sort Choice of algorithm - Merge sort is generally faster to sort lists, so would be the recommended algorithm
Flowchart Symbols	

<table>
<tr><td>Data Types</td><td>

Data type	Description	Example
INTEGER	A whole number	`1475`, `0`, `-5`
REAL	A number with a decimal point	`56.75`, `6.0`, `-2.456`, `0.0`
BOOLEAN	Either TRUE or FALSE	`TRUE`, `FALSE`
CHARACTER	A single alphabetic or numeric character	`'a'`, `'K'`, `'4'`, `'@'`, `'%'`
STRING	A sequence of one or more characters	`"Jo Hobson"`, `"123"`

</td></tr>
<tr><td>Boolean operators and programming symbols</td><td>

Symbol / keyword	Meaning	Symbol / keyword	Meaning
`<`	Less than	`+`	Concatenation
`<=`	Less than or equal to	`if elseif else`	Branch depending on condition
`>`	Greater than	`switch case default`	Branch depending on case
`>=`	Greater than or equal to	`input()`	Get user input
`==`	Equal to	`print()`	Output to the user
`=`	Assignment	`for`	Repeat a set number of times
`!=`	Not equal to	`while`	Repeat while a condition is true
`*`	Multiply	`do until`	Do a loop until a condition is true
`^`	Exponent	`str()`	Convert to a string
`+`	Addition	`int()`	Convert to an integer

</td></tr>
<tr><td>Trace Tables</td><td>Trace tables are used to help find errors in a program.
Variable names and outputs are put in columns.
The programmer traces through the program line by line. updating the values of variables and outputs.
A row is used for each iteration.</td></tr>
<tr><td>Syntax error</td><td>An error casuesd by not following the rules of the language e.g missing brackets or quotemarks.</td></tr>
<tr><td>Logical error</td><td>The logic of the program is incorrect – e.g. wrong values used to create a total.</td></tr>
</table>