

Y10 Computer Science – Autumn 1 – Algorithms & Python programming



Section A: Key vocabulary	
Decomposition	Breaking a problem down into smaller parts so that you can create a solution.
Algorithm	A list of instructions that will do something.
Selection	A structure in programming that enables you to do different things depending on if a condition is met or not. IF
Iteration	A structure in programming that enables you to repeat something. FOR, WHILE
Variable	A location that is given an identifier that stores data. The data can be changed.
Assignment	The process of giving data to a variable.
Pseudocode	A language similar to English that is used for writing algorithms for planning (not programs).
Abstraction	Looking at a problem and ignoring the irrelevant information.
Definite iteration	Looping that you know will end. This is usually a For loop but can be a while if it is designed with a counter to track the number of loops.
Indefinite iteration	Looping that you don't know when it will end as it is dependent on a condition to be met.
Condition	A criteria that has to be met for something to happen, used in selection and indefinite iteration. For example a = 3.
Data type	Limits what can be stored in a variable. These include: string, integer, float, Boolean, character.

Section B: Algorithms
<p>Linear search Searching through a list of data item by item from the start to the end or until you find the item you want. Data doesn't need to be sorted.</p> <p>Binary search Searching for an item in a sorted list of data by eliminating data that is not in the half the value should be in. Repeat this process until you have the value.</p> <p>Merge sort Splitting a list of values in half, then half again and repeat until you have them in single items. Then look at the data in pairs and merge them back together in the right order then merge pairs and order, repeat merge and sort until all the data is sorted.</p> <p>Bubble sort Sorting data in pairs comparing the data and swapping place if needed. Then looking at the next pair until reaching the end. Repeat this until a full check is completed without swaps.</p> <p>When solving a problem you will need to do the following:</p> <ul style="list-style-type: none"> Identify a problem Abstract irrelevant information Decompose the problem into chunks Decompose each chunk into smaller parts Develop a program for each part Test the parts and fix errors Test the whole program and fix errors Evaluate the program Redevelop based on evaluation

Section C: Programming Commands	
Output procedure	print()
Output a string	print("hello")
Output stored data	print (age)
Input procedure	input("instruction")
Data needs storing in a variable	inputs default to strings
Variable declaration and assignment	age = 40 age = input("Enter age: ")
Data Types and casting (changing data from one type to another)	String = "hello" str() Integer = 78 int() Float = 76.5 float() Boolean = True or False
Selection Indents matter	if age < 13: print("No account") else: print("Yes account")
Iteration Indents matter	for loop in range(1:10): print(2*loop) while age <13: print("No account")
Useful website for more examples: www.w3schools.com/python/	

Y10 Computer Science – Autumn 2 – Python Programming Continued



Section A: Key vocabulary

Subroutine	A section of code that is coded out of sequence that can be called at any time.
Function	A subroutine that returns data back to the global pool of variables.
Procedure	A subroutine that does not return any data to the global pool of variables.
Parameter	The data that is passed into a subroutine (both functions and procedures can have parameters).
Array	A list of data stored as a single variable. Data can be; added to it, deleted from it, sorted in order, edited and extracted.
Index	The position of data stored in a list. Indexing starts at 0.
Logical operators	Mathematical symbols used in conditions. >, <, >=, <=, ==, !=
Boolean operators	Keywords used in conditions which result in a True or False response. AND, OR, NOT
Global variable	Variables that can be accessed any where in the program and are declared at the start of a program.
Local variable	Variables that are declared in a subroutine and can only be used in that subroutine.
Return	The command that returns data from a function to a global variable.

Section B: Robust programming

Program structure

As programs may be read by different people in the development and redevelopment stages it is essential to make the program Robust. To do this you just need to follow these simple steps in order (add to these sections as you create the code keeping your code organised):

1. Import libraries/modules
2. Declare global variable
3. Create subroutines
4. Create the main program

Comments

All sections should have suitable comments using either the # for short comments or """ for extended comments""".

Validation

When ever data is inputted by a user this should be checked for validity to stop human error. For example if you are expecting 'Yes' or 'No' the program should only accept 'Yes' or 'No'. We usually use while loops with Boolean and logical operators to complete this.

Section C: Programming Commands

Subroutine procedure	<pre>def procedureName(parameters): code in the procedure Example: name = "Joe" def outputName (name): print(name) #calling procedure outputName (name)</pre>
Subroutine function	<pre>def functionName(parameters): code in the procedure return parameter/s Example: name = 0 firstname = "Joe" surname = "Bloggs" def Name (firstname, surname): name = firstname + surname return name #calling function name = Name (firstname, surname)</pre>
Array	<pre>letters = ["a", "b", "c", "d"] print(letters) print(letters[1]) print(letters[0:2])</pre>
Useful website for more examples: www.w3schools.com/python/	