

AS & A LEVEL Topic Exploration Pack

H046/H446

COMPUTER SCIENCE

Theme: Programming Techniques

attp://w

TAXABLE PARTY OF

September 2015







We will inform centres about any changes to the specification. We will also publish changes on our website. The latest version of our specification will always be the one on our website (<u>www.ocr.org.uk</u>) and this may differ from printed versions.

Copyright © 2015 OCR. All rights reserved.

Copyright

OCR retains the copyright on all its publications, including the specifications. However, registered centres for OCR are permitted to copy material from this specification booklet for their own internal use.

Oxford Cambridge and RSA Examinations is a Company Limited by Guarantee. Registered in England. Registered company number 3484466.

Registered office: 1 Hills Road Cambridge CB1 2EU

OCR is an exempt charity.

Contents

| Contents | 3 |
|------------------------|---|
| Programming Techniques | 4 |
| Task 1 | 4 |
| Task 2 | 5 |
| Task 3 | 5 |

This Topic Exploration Pack should accompany the OCR resource 'Programming Techniques' learner activities, which you can download from the OCR website.



This activity offers an opportunity for English skills development.



Programming Techniques

This subject introduces the concepts of procedural programming and the appropriate terminology to be used when teaching.

Keywords to be able to explain:

Modularity, variable, procedure, scope, function, sub-routine, constant, argument, global variable, parameter, local variable, data type.

Task 1

Look up definitions for these keywords and group them under four different headings, copy and paste the definition and then rewrite it in your own words.

Programming constructs are the tools used to create a programme and for procedural programming there are three main types.

Sequence: One instruction follows another in a successive manner.

Branching: Usually known as a conditional statement the most common of these is the if-then-else statement.

Iteration: Is repetition of a sequence of tasks for a specified amount of times or, more often, until a certain condition has been met.

If students have no prior knowledge of any programming then a brief explanation of what a variable is will be necessary from the beginning.

Groups for Task 1

| Blocks of code | Passing data | Who can see what | What am I? |
|----------------|--------------|------------------|------------|
| Modularity | Parameter | Local variable | Variable |
| Function | argument | Global variable | Constant |
| Procedure | | Scope | Data type |
| Subroutine | | | |

Modularity has slightly different definitions for procedural and OOP, extra marks for someone who finds both and can differentiate between the two.

Descriptions are the same as the specification but are in a different order so that it flows and builds up to a natural progression.



Task 2 - Basic programming tasks

From a procedural language point of view modularity is used to break a program down into manageable parts that can be self-contained and tested independently, they can also be written by two or more programmers.

Variables and constants: In programming it is sometimes necessary to use variables that need to be used throughout the whole program, these are declared in the main body of the program and they are called global variables. Local variables are declared within a function/procedure and are only used within that module.

Parameters: When parameters/arguments are passed to a procedure/function they can be passed in two ways: One is by value, where the value to be used is given as an actual value and is discarded at the end of the module. The other is to pass a reference to an address where the value is stored, once the module has completed its task any value changed in the variable will be kept.

This is normally referred to as parameter passing by value or by reference.

Task 2 programs can be checked to see if they perform as required. Good habit to get them saving different versions of files.

Recursion

In the previous section it was stated that an efficient way to solve a problem was to use an iterative approach to programming and, rather than repeating lines of code, you could 'loop' the sequence required. The same principle applies to recursion.

Task 3

Extend Square_4 to run from a procedure (no need to use parameters yet, just put the whole thing in a procedure). Save it as Square_7.

Using Square_4 again, create a procedure to just calculate the square root and use parameters to pass the values into and out of that procedure. Save it as Square_8.

Create an algorithm to change this to a recursive solution.

Discuss the differences between iteration and recursion and where it would be appropriate to use either.

It may be a bit of a stretch for them to do this but should give a good understanding of parameters, depends on level of ability.





We'd like to know your view on the resources we produce. By clicking on the 'Like' or 'Dislike' button you can help us to ensure that our resources work for you. When the email template pops up please add additional comments if you wish and then just click 'Send'. Thank you.

If you do not currently offer this OCR qualification but would like to do so, please complete the Expression of Interest Form which can be found here: <u>www.ocr.org.uk/expression-of-interest</u>

OCR Resources: the small print

OCR's resources are provided to support the teaching of OCR specifications, but in no way constitute an endorsed teaching method that is required by the Board, and the decision to use them lies with the individual teacher. Whilst every effort is made to ensure the accuracy of the content, OCR cannot be held responsible for any errors or omissions within these resources.

© OCR 2015 - This resource may be freely copied and distributed, as long as the OCR logo and this message remain intact and OCR is acknowledged as the originator of this work. OCR acknowledges the use of the following content: Thumbs up and down icons: alexwhite/Shutterstock.com

Please get in touch if you want to discuss the accessibility of resources we offer to support delivery of our qualifications: resources.feedback@ocr.org.uk



OCR customer contact centre

General qualifications Telephone 01223 553998 Facsimile 01223 552627 Email general.qualifications@ocr.org.uk



For staft training purposes and as part of our quality assurance programme your call may be recorded or monitored. ©OCR 2015 Oxford Cambridge and RSA Examinations is a Company Limited by Guarantee. Registered in England. Registered office 1 Hills Road, Cambridge CB1 2EU. Registered company number 3484466. OCR is an exempt charity.