

Computer Science (OCR) - Mr Howell - July 2018

AS OCR H046 2015/16 & 2016/17 A2 OCR H447 2016/17

Reading List

- 1 *OCR A Level Computer Science* by G Rouse, J Pitt & S O'Byrne (Hodder)

This book covers the whole scheme of work on the A level course.

- 2 Wiki A level Computing website

https://en.wikibooks.org/wiki/A-level_Computing

There are some topic areas that are explained well with suitable examples

- 3 www.codecademy.com

Learn how to program in Python, JavaScript, HTML and CSS. This is part of your course where you will need to be confident at programming in the given languages

- 4 www.teach-ict.com

Excellent place to get notes and information about many topics in the course

- 5 <http://stackoverflow.com/>

Great place to ask for help on programming concept. Excellent forum where you can ask questions about any topic area

10 programming challenges that you could use python or JavaScript to complete.

Challenge	
Factorial Finder	1 The Factorial of a positive integer, n, is defined as the product of the sequence n, n-1, n-2, ...1 and the factorial of zero, 0, is defined as being 1. Solve this using both loops and recursion.
Speed Tracker	<p>Create a program that takes a time for a car going past a speed camera, the time going past the next one and the distance between them to calculate the average speed for the car in mph. The cameras are one mile apart.</p> <p>Extensions:</p> <p>1. Speed cameras know the timings of each car going past, through number plate recognition. Valid number plates are two letters, two numbers and three letters afterwards, for example XX77 787.</p> <p>2 Produce a part of the program that checks whether a number plate matches the given pattern. Tell the user either way.</p> <p>2. Create a program for creating a file of details for vehicles exceeding the speed limit set for a section of road. You will need to create a suitable file with test data, including randomised number plates and times. You will then use the code you've already written to process this list to determine who is breaking the speed limit (70mph) and who has invalid number plates.</p>
Thief!	<p>A thief has managed to find out the four digits for an online PIN code, but doesn't know the correct sequence needed to hack into the account.</p> <p>3 Design and write a program that displays all the possible combinations for any four numerical digits entered by the user. The program should avoid displaying the same combination more than once.</p> <p>Submit a fully detailed Showcase for your program.</p>
Classification	<p>A simple classification system asks a series of Yes/No questions in order to work out what type of animal is being looked at.</p> <p>Eg Does it have 4 legs? Does it eat meat? Does it have stripes?</p> <p>These systems can often be drawn using a "tree" structure. Carry out some simple research on classification trees, then write a program to help the user decide between the following:</p> <p>4 horse, cow, sheep, pig, dog, cat, lion, tiger, whale, dolphin, seal, penguin, ostrich, sparrow, spider, ant, bee, wasp, termite, octopus, squid</p> <p>Is there a better way to do this than using 101 IF...ELSE...END IFs?</p> <p>Develop your classification system for your own area of interest: pop bands; pokemon; cars; footballers; teachers; diseases etc.</p>
Fruit Machine	<p>Write a program to simulate a Fruit Machine that displays three symbols at random from Cherry, Bell, Lemon, Orange, Star, Skull.</p> <p>5 The player starts with £1 credit, with each go costing 20p. If the Fruit Machine "rolls" two of the same symbol, the user wins 50p. The player wins £1 for three of the same and £5 for 3 Bells. The player loses £1 if two skulls are rolled and all of his/her money if three skulls are rolled. The player can choose to quit with the winnings after each roll or keep playing until there is no money left.</p>
Unit Converter (temp, currency, volume)	

6	<p>Converts various units between one another. The user enters the type of unit being entered, the type of unit they want to convert to and then the value. The program will then make the conversion.</p>
7	<p>Credit Card Validator Takes in a credit card number from a common credit card vendor (Visa, MasterCard, American Express, Discoverer) and validates it to make sure that it is a valid number (look into how credit cards use a checksum).</p>
8	<p>Arithmetic test A primary school teacher wants a computer program to test the basic arithmetic skills of her students. Generate random questions (2 numbers only) consisting of addition, subtraction, multiplication and division. The system should ask the student's name and then ask ten questions. The program should feed back if the answers are correct or not, and then generate a final score at the end.</p> <p>Extensions:</p> <ol style="list-style-type: none"> 1. Extend your program so that it stores the results somewhere. The teacher has three classes, so you need to enable the program to distinguish between them. 2. The teacher wants to be able to log student performance in these tests. The teacher would like the program to store the last three scores for each student and to be able to output the results in alphabetical order with the student's highest score first out of the three.
9	<p>Happy Numbers A happy number is defined by the following process: Starting with any positive integer, replace the number by the sum of the squares of its digits, and repeat the process until the number equals 1 (where it will stay), or it loops endlessly in a cycle which does not include 1. Those numbers for which this process ends in 1 are happy numbers, while those that do not end in 1 are unhappy numbers. Display an example of your output here. Find the first eight happy numbers.</p>
10	<p>Number Names Show how to spell out a number in English. You can use a pre-existing implementation or make your own, but you should support inputs up to at least one million (or the maximum value of your language's default bounded integer type, if that's less).</p> <p>Extensions:</p> <ol style="list-style-type: none"> 1. Create support for inputs other than positive integers (like zero, negative integers, and floating-point numbers).