



GCSE Computer Science: Effective Teaching and Learning for Paper 2

Welcome to this Professional Development Training

This free online event supports teachers delivering Pearson Edexcel GCSE Computer Science whether you are new or have been teaching this course for five years. It is part of a series of connected webinars this school year.

The training covers a variety of approaches for the effective teaching of Paper 2 and how to provide feedback to students on their progress.

You will:

- be shown options for balancing the theory and programming elements of the course
 - see how to organise the existing Pearson CT (Computational Thinking) units with free online platforms and resources
 - be shown practical approaches for T&L of programming
 - consider forms of assessment and feedback in both Years 10 and 11.
-

Questions

Please answer the following poll questions when they are launched:

- When was / is your first Pearson GCSE Computer Science cohort exam?
- What platform does your school use?

Agenda

Effective Teaching and Learning for Paper 2, 60mins

In this session we are going to:

- **share the different contexts** that we have in our schools
- share the **key documents** you should know fully
- consider the best **balance for teaching theory and programming** although this may vary between classes
- look at resources to teach programming starting with the **free Pearson Edexcel units** but also ways to enhance with free **additional resources**
- look at **practical approaches for Teaching and Learning programming** including the recent **feedback from the examiners**
- show examples of **assessment in Year 10 and Year 11** – it might vary...
- **future webinar dates and Subject Advisor contact.**



Questions

Please answer the following poll questions when they are launched:

- Which IDE(s) do you use in the exam?
- Which IDE(s) do you use for regular teaching?

Key documents for Paper 2

GCSE (9–1) Computer Science Specification (p5, 13–17, –) – [link](#)

Programming Language Subset (PLS) Booklet (v6 / coloured paper) – [link](#)*

Pearson Edexcel's Good Programming Practice Guide (Issue 1.2) – [link](#)*

Instructions for Conduct of the Examination (ICE 1.9) – [link](#)*

Pearson Edexcel's Getting Started Guide (Issue 3, p5–8, 9, 26 – Topic 6) – [link](#)*

Specimen Exam papers / SAMs 1–4 / 2022–2025 Exam past papers – [link](#)

**Always check you have the latest versions.*

What is your balance of teaching theory and practical?

Questions

Please answer the following poll questions when they are launched:

- What is your **curriculum time allocation *per fortnight*** in hours?
- How do you **balance your lesson time for both theory and programming ?**

A1. Programming must be done...

Page 16 of the specification reminds us that we all have to do some programming...

... and the Practical Programming Statement needs to be signed by SLT and sent by 31st May each year.

Practical Programming Statement (PPS)

Centres are required to complete a Practical Programming Statement (PPS), which confirms they have taken reasonable steps to ensure that each student sitting our GCSE in Computer Science has had the opportunity to demonstrate their programming skills in terms of design, write, test and refine programs in Python during their course of study. The PPS should be completed by a member of the senior leadership team at the centre. Failure by a centre to provide a completed PPS to us in a timely manner will result in potential malpractice and/or maladministration.

The PPS will be made available by the 31st of January in the year of the examination on the [GCSE in Computer Science section of our website](#).

The completed PPS must be submitted to us by the 31st of May in the year of the examination. Centres are reminded that if they do not submit this form it may result in malpractice/maladministration.

A2a. Balance – WAGOLL 1

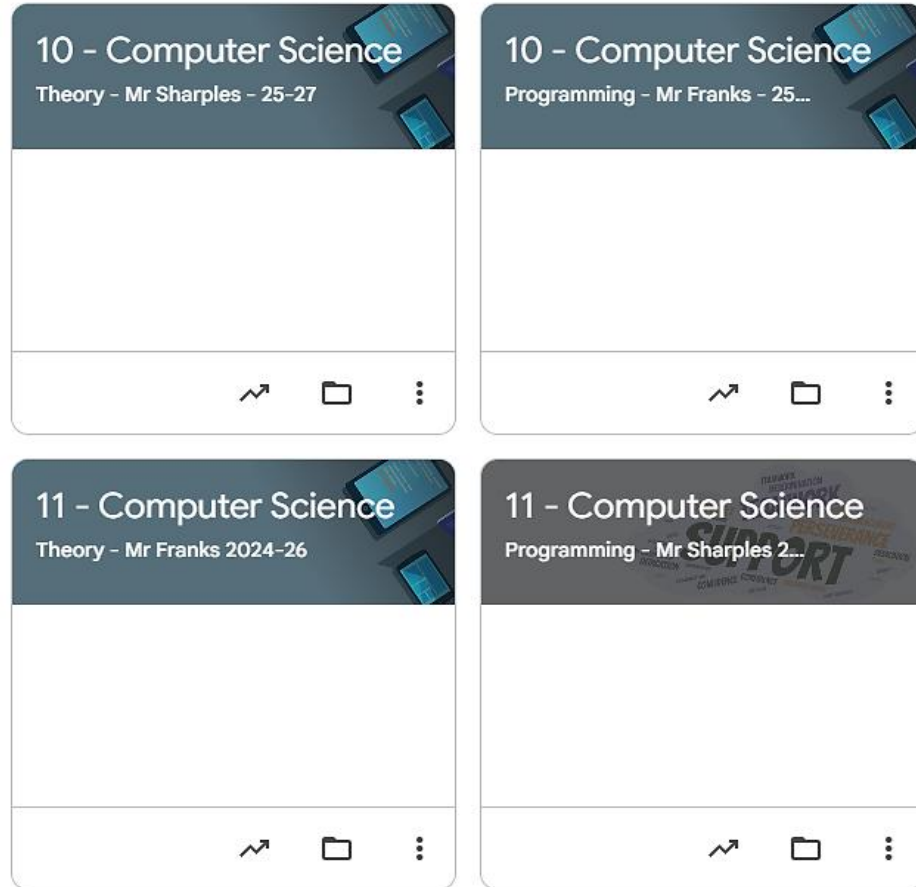
Block 6	Y10_06_P	Y10-06-PSum1	P31	LANs and WANs	Give reasons why computers are connected on a network Differentiate between a LAN and a WAN Categorise tasks according to the type of network used to carry them out Explain the benefits to organisations of a WAN Explain why protocols are needed on a network Describe the purpose of an IP address	4.1.1 4.1.2 4.1.3 4.1.6	29.04.2025
			P32	Network speed	Define the meanings of the terms 'bandwidth' and 'latency' Explain how bandwidth and latency affect the performance of a network Use bits per second (bps) to describe network speed Construct expressions involving file size, transmission rate and time	2.3.1 4.1.5	01.05.2025
			P33	Connectivity	Differentiate between wired and wireless connectivity Explain how data are transmitted along copper and fibre-optic cables Compare the performance of copper and fibre-optic cables and give examples of their use Describe how high-speed broadband is delivered	4.1.4	13.05.2025
			P34	Wired vs wireless	Describe how devices are connected on a wireless network Compare the performance of wired and wireless LANs and give examples of situations where one is preferable to the other Summarise the characteristic of Wi-Fi, Bluetooth, RFID, Zigbee and NFC and give examples of their use	4.1.4	15.05.2025
			P35	Network topologies	Define the term 'topology' Describes the characteristics of bus, star and mesh network topologies Draw and label a diagram of each topology Match descriptions to network topologies Match descriptions of what they do to internet components (backbone, POP, NAP, router)	4.1.3 4.1.8	03.06.2025
			P36	Assessment	Give three reasons for connecting devices in a network Explain how a LAN differs from a WAN Define the term 'internet backbone' Describe the function of a router Explain how data are transmitted on a fibre-optic cable State two advantages and two disadvantages of using wireless to connect devices on a LAN rather than cable Construct an expression to calculate the time needed to transmit a file over a network Explain why protocols are needed on a network	4.1.1 4.1.2 4.1.3 4.1.4 4.1.5 4.1.6	05.06.2025

A2b. Balance – WAGOLL 1

Block	Section	Subfolder	Lesson number	Lesson title	What students will learn	Spec. ref.	Date
Block 6	Y10_06_CT	Y10-06-CTSum1	CT31	Turtle introduction, pens and lines	Decompose a problem Use turtle graphics to draw lines	6.1.1 6.1.2 6.1.4	24.04.25
			CT32	Turtle movement, coordinates, polygons, subprograms	Use Cartesian coordinates Incorporate selection, repetition, and iteration into turtle graphics Use subprograms	6.2.2 6.3.1 6.3.2 6.4.1 6.6.2	24.04.25
			CT33	Turtle pens, colours, filling, and circle	Use turtle pens of different colours Use turtle pens of different sizes Use turtle fill in closed shapes	6.2.2 6.3.1 6.3.2 6.6.1 6.6.2	08.05.2025
			CT34	Turtle, combining subprograms, layers	Combine subprograms to produce a turtle graphics image	6.2.2 6.3.1 6.3.2 6.6.1 6.6.2	08.05.2025
			CT35	Turtle big problem	Decompose a problem into smaller parts Combine subprograms to create a solution	1.1.1 6.1.1 6.1.2 6.4.1 6.6.1 6.6.2	22.05.2025
			CT36	Assessment	Use the turtle module, programming constructs, and subprograms to create images	6.2.2 6.3.1 6.3.2 6.6.1 6.6.2	12.06.2025

A3a. Balance – WAGOLL 2 Using Google Classrooms

Separate Google Classrooms for theory and programming through Years 10 and 11.



Pearson Edexcel units 10-3,4,5,6

	10-6 Python Turtle - Mr Sharples ill - sor...	Edited 14 Oct	
	Reminder Slides 10-5 Merge Sort, Data F...	Posted 7 Oct	
	Assessment 10-5 run through	Edited 7 Oct	
	Notes about the 10-5 Assessment	Edited 25 Jun	
	10_05 Activity solutions	Posted 24 Jun	
	10_05 Homework solutions	Posted 24 Jun	
	Answers to trinkets for Y10-04-CT22 Lin...	Posted 2 Apr	
	Debugging with Thonny notes	Posted 3 Feb	
	Answers for Edexcel Trinket Activities	Edited 21 Feb	
	Edexcel Trinkets - Spring term	Posted 29 Jan	

A3b. Balance – WAGOLL 2 in detail

1 lesson (55m) Theory per week, 2 lessons (a double – 1h 50m) Programming per week:

	2024-26	Mr Franks	Mr Sharples	Mr Sharples
	Blank Class	Mon p7 Theory	Tues p1 Programming	Tues p2 Programming
01/09/2025				
08/09/2025		5.1 - Environmental issues - PGO slides	GPPG Booklet 1	Revision
15/09/2025		Smart Revise - reminder	Topic test 10-5 Files and Authentication + Qu6	Topic test 10-5 Files and Authentication + Qu6
22/09/2025		GPPG Booklet 2 / SR time	Go through reading files	Go through reading files
29/09/2025		5.3 Legislation and privacy RPI lesson & PGO slides	Go through writing files	Go through writing files
06/10/2025		GPPG Booklet 3 / SR time	Go through authentication	Go through authentication
13/10/2025		GPPG Booklet 4 / SR time	T2Code 10 - Turtle	T2Code 10 - Turtle
20/10/2025	Halfterm	Revision / SR time	Revision / SR time	Revision / SR time
27/10/2025	Halfterm	Revision / SR time	Revision / SR time	Revision / SR time
03/11/2025		<i>Inset day</i>	Mocks	Mocks
10/11/2025		Mocks	Mocks CS Paper 1 - SAM04	Mocks CS Paper 1 - SAM04

A4. Analysis –

How do you know if the balance of teaching Theory: Programming works for your class or classes?

Pearson have looked internally every year to see if there is any significant difference across the whole cohort between P1 and P2 and they do not see any significant difference in outcomes.

However, you can do your own analysis each year to see if there is a difference:

I think there is a clear pattern for the majority of students that 1:2 Theory: Programming time in lessons results in better programmers...

Sample selection from real 2025 results:

P	Q	R	S	T	U	V	W
	Paper 1 /75		Paper 2 /75		residual to show difference in performance in theory of programming	Total both /150	Total both %
pupil 1	58	77%	75	100%	-23%	133	88.67
pupil 2	7	9%	8	11%	-1%	15	10.00
pupil 3	46	61%	38	51%	11%	84	56.00
pupil 4	27	36%	31	41%	-5%	58	38.67
pupil 5	31	41%	64	85%	-44%	95	63.33
pupil 6	27	36%	34	45%	-9%	61	40.67
pupil 7	52	69%	65	87%	-17%	117	78.00
pupil 8	28	37%	32	43%	-5%	60	40.00
Averages:		46%	43	58%	-12%	43	28.92

Column U shows the comparison in results /75 for P1 and P2. Negative value favours programming.

A5. Other considerations

Any of these may be relevant to your balancing decision:

- Ability profile of students
- Motivation
- Time of day
- Availability of teachers
- Prior experience of teacher AND students
- Teacher led / independent working / paired working
- The progress of each class

Resources to teach programming

Bla.
Resources to
teach
programming


Scheme of work - Learning content mapping


The materials of our Scheme of Work have been classified into blocks, from 1 to 10.
You can download the materials for each block individually as a zip folder.


	Term1		Term 2		Term 3	
	Half Term 1	Half Term 2	Half Term 1	Half Term 2	Half Term 1	Half Term 2
Year 10	Block 1	Block 2	Block 3	Block 4	Block 5	Block 6
Year 11	Block 7	Block 8	Block 9	Block 10		




Scheme of work

 [GCSE \(9-1\) Computer Science Scheme of Work Guidance](#)
| PDF 1.3 MB | 07 June 2024

 GCSE (9-1) Computer Science Introduction to the Scheme of Work
| PDF 1.0 MB | 05 June 2024


 GCSE (9-1) Computer Science Scheme of Work - Learning Content Mapping
| XLSX 367.0 KB | 05 June 2024


 GCSE (9-1) Computer Science Scheme of Work Block 1
| ZIP 11.6 MB | 05 June 2024

Download
content for
each of the
10 blocks


B1b. Resources to teach programming


Lesson number	Lesson title	What students will learn	Spec. ref.
CT31	Turtle introduction, pens and lines	Decompose a problem Use turtle graphics to draw lines	6.1.1 6.1.2 6.1.4
CT32	Turtle movement, coordinates, polygons, subprograms	Use Cartesian coordinates Incorporate selection, repetition, and iteration into turtle graphics Use subprograms	6.2.2 6.3.1 6.3.2 6.4.1 6.6.2
CT33	Turtle pens, colours, filling, and circle	Use turtle pens of different colours Use turtle pens of different sizes Use turtle fill in closed shapes	6.2.2 6.3.1 6.3.2 6.6.1 6.6.2
CT34	Turtle, combining subprograms, layers	Combine subprograms to produce a turtle graphics image	6.2.2 6.3.1 6.3.2 6.6.1 6.6.2
CT35	Turtle big problem	Decompose a problem into smaller parts Combine subprograms to create a solution	1.1.1 6.1.1 6.1.2 6.4.1 6.6.1 6.6.2
CT36	Assessment	Use the turtle module, programming constructs, and subprograms to create images	6.2.2 6.3.1 6.3.2 6.6.1 6.6.2


 CT33_Code_Snippets


 CT33_Slide_Code_Snippets


 Y10_06_CT33_Activities

 Y10_06_CT33_Activities_Solutions

 Y10_06_CT33_Homework

 Y10_06_CT33_Homework_Solutions

 Y10_06_CT33_LessonPlan

 Y10_06_CT33_Slides

B1c. Resources to teach programming

Activity 2

Here is a Python program.

```
# Program to calculate the volume of a cone
PI = 3.141 # create a constant for the value of PI
radius = 5
print("Volume of the cone is", volume)
volume = PI * radiusSquared * (height/3)
radiusSquared = radius ** 2
height = 8
```

- Copy and paste the program into your Integrated Development Environment (IDE) and run it.

- Re-order the lines of code so that the program runs without error.

.....

.....

.....

.....

.....


.....


.....


Describe what happens when you run the program provided.


[Pearson's Good Programming Practice Guide](#)


Bld. Resources to teach programming


 Y10-01-CT2_Activity_Ext
Due Fri 12 September 2025 at 23:27 • 9 Submissions


 Y10-01-CT2_Activity2
Due Fri 12 September 2025 at 23:27 • 11 Submissions

 Y10-01-CT1_Activity_Ext
Due Fri 12 September 2025 at 23:26 • 10 Submissions

 Y10-01-CT1_Activity3
Due Fri 12 September 2025 at 23:26 • 12 Submissions

 Y10-01-CT1_Activity4
Due Fri 12 September 2025 at 23:26 • 11 Submissions

 Y10-01-CT1_Activity1-2
Due Fri 12 September 2025 at 23:25 • 12 Submissions


 Y10-01-CT2_Activity2

```
1 # Program to calculate the volume of a cone
2 PI = 3.141 # create a constant for the value of PI
3 radius = 5
4 print("Volume of the cone is", volume)
5 volume = PI * radiusSquared * (height / 3)
6 radiusSquared = radius ** 2
7 height = 8
8
```

✓ Submit

▶ Run

Assignment Output

 Y10-01-CT2_Activity2
Due Fri 12 September 2025 at 23:27

Activity 2

Here is a Python program.

Run the program - what happens?

Re-order the lines of code so that the program runs without error **and** outputs the correct answer.

B2a.

Resources to teach programming – Trinket Connect

trinket Connect

Home

Plans Learn Help csharplesadmin Sign Out

/ @RS Pearson Edexcel Y10/11 Trinkets THIS ONE 2025-26 by Chris Sharples

Dashboard Course Users < >

Y10-04-CT24: Assessment Qu6*

Y10-04-CT24: Assessment Qu8*

Y10-04-CT24: Assessment Qu9 (PGO2A Qu...

Page Assignment

10-05

Y10-05-CT25: Merge sort MISS OUT 2025

Y10-05-CT26: Reading files*

Y10-05-CT27: String processing*

Y10-05-CT28: Writing files*

Y10-05-CT29: Authentication*

Y10-05-CT26272829Homework answers

Y10-05-CT30: Assessment Qu3

Y10-05-CT30: Assessment Qu4

Y10-05-CT30: Assessment Qu5

Page Assignment

10-06

Y10-06-CT31: Turtle introduction, pens an...

Y10-06-CT32: Turtle movement, coordinat...

Y10-06-CT33: Turtle pens, colours, filling a...

Y10-06-CT34: Turtle combining subprogra...

Y10-06-CT35: Turtle big problem

Y10-06-CT36: Assessment02

Y10-06-CT36: Assessment03

Topic Outline

Y10-05-CT26: Reading files*

Click [here](#) for Slides for Y10-05-CT26: Reading files

Worksheet Activities followed by Homework Tasks.

Y10_05_CT26_Activities:

Activity 1a - Reading and just printing data from a file

A program has been started that loads data from a file containing records. Use slide 10 to read the data into your python code and print it out.

main.py CT26 Activity1.txt

```
1 #Y10_05_26_A1Remix2025
2
3 # -----
4 # Constants
5 # -----
6 FILENAME = "CT26_Activity1.txt"
7
8 # -----
9 # Global variables
10 # -----
11 usersTable = [] # Holds all the records
12
13 # -----
14 # Subprograms
15 # -----
16 def loadData ():
17
18     # =====> Open the file for reading only
19     theFile =
20
21     # Use the example code in Slide 10 to output out the data
22
23     # =====> Close the file
```

Result

Instructions

A program has been started that loads data from a file containing records.

Use slide 10 to read the data into your python code and print it out.

main.py CT26 Activity1.txt

```
1 #Y10_05_26_A1Remix2025b-
2 # Constants
3 # -----
4 FILENAME = "CT26_Activity1.txt"
5
6 # -----
7 # Global variables
```

Result

Instructions

A program has been started that loads data from a file containing records.

Amend the code to complete the program.

Copyright 2025 Pearson. All rights reserved.

21

B2b. Time 2 Code – Craig’n’Dave – Free



TIME2CODE

Q Search

✓ I can do this

Home > Python Course

Python course

Level 1
Inputs, outputs and structure
An introduction to inputs, outputs and structured programs.
[START LEVEL 1 →](#)

Level 2
Selections
Learn to add logic to your programs using the concept of selection.
[START LEVEL 2 →](#)

Level 3
Condition controlled repetitions
An introduction to condition-controlled iterations using while loops.
[START LEVEL 3 →](#)

Assessment point
Levels 1-3
Test yourself on everything you have learned so far.
[START ASSESSMENT →](#)

Level 4
Number operations and libraries
A mathematical masterclass in Python, and some more advanced string formatting.
[START LEVEL 4 →](#)

Level 5
One-dimension arrays and lists
Learn how to use the structured list data type in your programs.
[START LEVEL 5 →](#)

Level 6
Count controlled iterations and scope
An introduction to count-controlled iterations using for loops.
[START LEVEL 6 →](#)

Assessment point
Levels 1-6

trinket Connect Home

★ Plans 📖 Learn 🆘 Help csharplesadmin Sign Out

@@RS Time 2 Code Unit Python Levels 1-10 THIS ONE 2025-26 by csharplesAdmin

[Dashboard](#) [Course](#) [Users](#) < >

Time 2 Code:

- Python level 1 – Inputs, outputs and structure
 - Assessment 1.3 Ball pit
- Python level 2 – Selections
 - Assessment 2.3 Seasons
- Python level 3 – Condition controlled iter...**
 - Assessment 3.3 Adder
 - Assessment point Levels 1-3
- Python level 4 – Number operations and libra...
- Assessment 4.3 Denary to binary
- Python level 5 – One-dimension arrays and lis...
- Assessment 5.3 Two-dice pig
- Python level 6 – Count controlled iterations an...
- Assessment 6.3 Notebook
- Python level 7 – String operations
 - Assessment 7.3 Notebook Caesar cipher
- Python level 8– Files and exceptions
 - Assessment 8.3 Notebook Caesar cipher
- Python level 10 – Turtle
 - Assessment 10.3 Artist

[Page](#) [Assignment](#)

Python level 3 – Condition controlled iterations

3.1 Capital city

main.py

#T2Unit03Task01

Result

Instructions

3.2 Car value

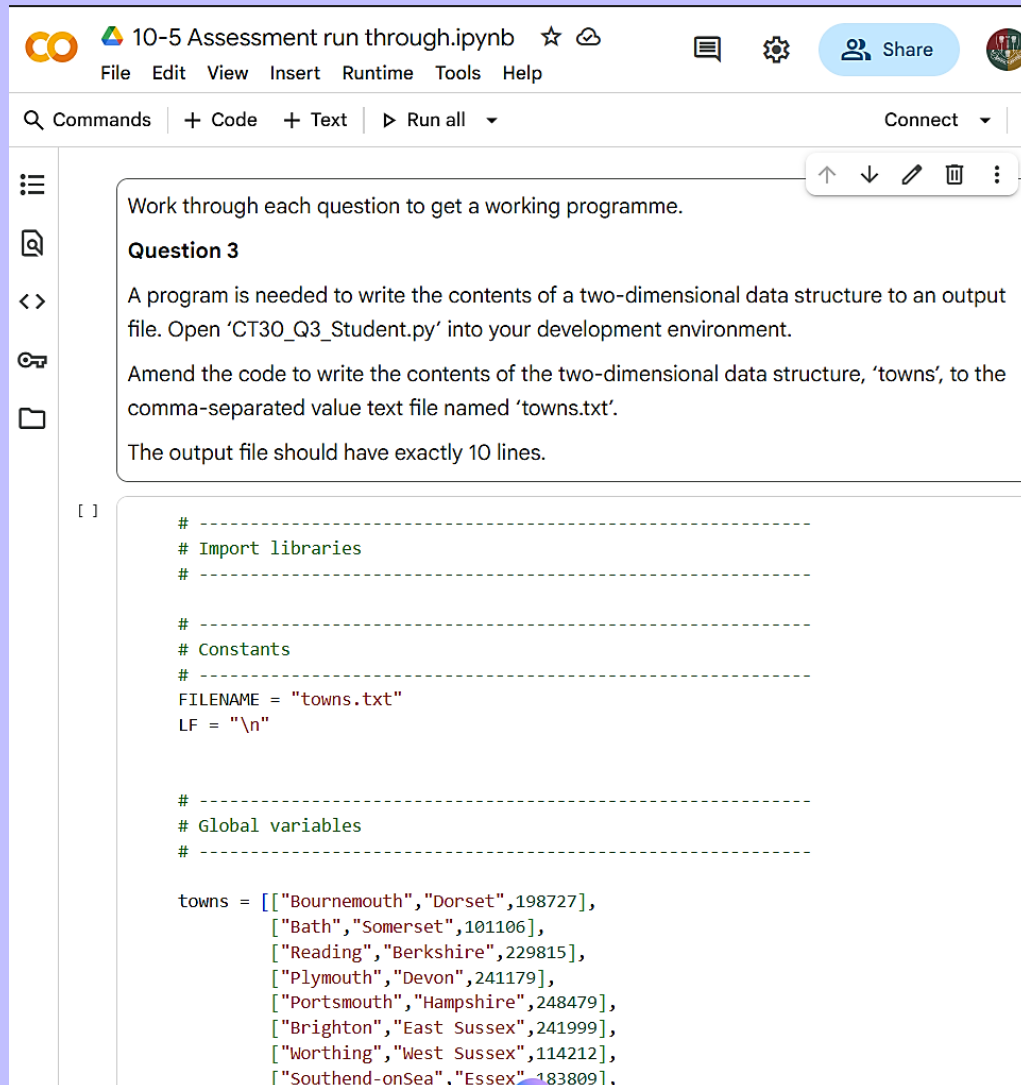
main.py

#T2Unit03Task02

Result

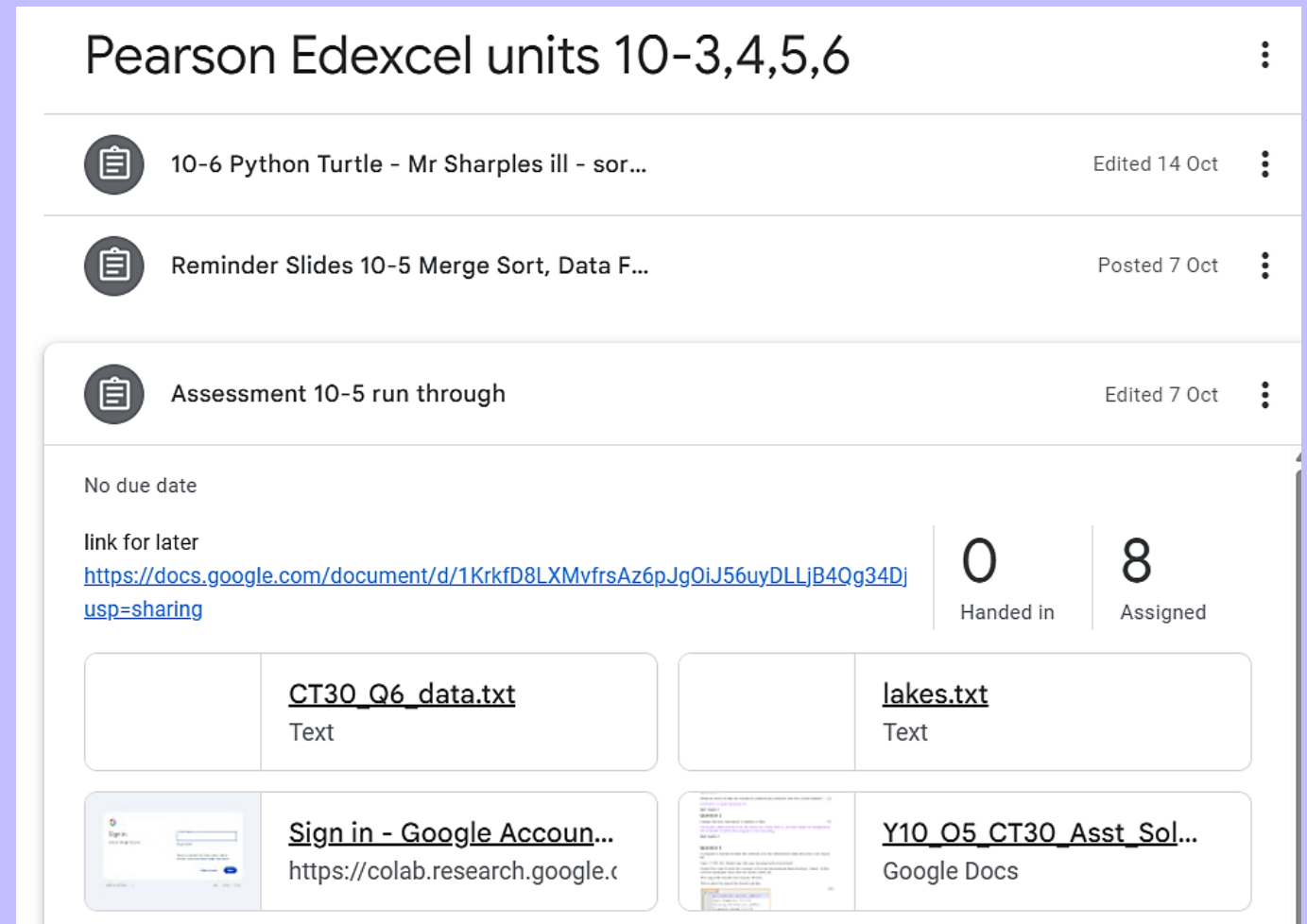
Instructions

B2c. Google Colab (integrated Jupyter notebooks)



The screenshot shows a Google Colab notebook interface. The top bar includes the Colab logo, the notebook title '10-5 Assessment run through.ipynb', and icons for file management, settings, and sharing. Below the top bar is a menu with 'Commands', '+ Code', '+ Text', and 'Run all'. The main content area is divided into two sections. The first section contains text instructions for 'Question 3', which asks the user to write a program to write the contents of a two-dimensional data structure to an output file named 'towns.txt'. The second section contains a Python code snippet that defines a list of towns and their populations, and writes this data to the 'towns.txt' file.

```
[ ]  
# -----  
# Import libraries  
# -----  
  
# -----  
# Constants  
# -----  
FILENAME = "towns.txt"  
LF = "\n"  
  
# -----  
# Global variables  
# -----  
  
towns = [ ["Bournemouth", "Dorset", 198727],  
          [ "Bath", "Somerset", 101106],  
          [ "Reading", "Berkshire", 229815],  
          [ "Plymouth", "Devon", 241179],  
          [ "Portsmouth", "Hampshire", 248479],  
          [ "Brighton", "East Sussex", 241999],  
          [ "Worthing", "West Sussex", 114212],  
          [ "Southend-on-Sea", "Essex", 183809],
```



The screenshot shows a Pearson Edexcel page for units 10-3,4,5,6. The page has a header with the title 'Pearson Edexcel units 10-3,4,5,6' and a vertical menu on the left. The main content area is divided into two sections. The first section contains a list of resources, including '10-6 Python Turtle - Mr Sharples ill - sor...', 'Reminder Slides 10-5 Merge Sort, Data F...', and 'Assessment 10-5 run through'. The second section contains a table with information about the assessment, including a link for later, a due date of 'No due date', and a table showing the number of items handed in (0) and assigned (8).

link for later	
https://docs.google.com/document/d/1KrkdD8LXMvfrsAz6pJgOiJ56uyDLLjB4Qg34Djusp=sharing	

No due date	
0	8
Handed in	Assigned

CT30_Q6_data.txt	
Text	

lakes.txt	
Text	

Sign in - Google Account...	
https://colab.research.google.com	

Y10_O5_CT30_Asst_Sol...	
Google Docs	

Summary of free resources to teach programming

Here is a list of the free resources mentioned:

- Pearson/Edexcel Computer Science – Scheme of Work resources – [link](#)
- Edublocks Classrooms – [link](#)
- Craig'n'Dave's Time 2 Code – [link](#)
- Google Colab (hosted Open Source Jupyter notebooks) – [link](#)
- Pete Dring's Blog with Code – [link](#)
- Trinket.io (Trinkets are free to set up but the editable course version (Connect) needs a licence) – [link](#)

Practical approaches for T&L of programming

C1. June 2025 Chief Examiner Findings

Paper 2 issues:

- Students struggle to run, test and debug programs effectively.
- Students fail to use *[read/write]* scaffolding and comments effectively.
- Many learners struggle with layout, indentation and sequencing logic.
- Students often gain partial functionality but fail to refine or optimise code.
- Students struggle to plan, break down and structure programs.

Practical Strategies for Teaching Programming (Part 1)

Building confident, independent coders:

- Use PRIMM or Use–Modify–Create frameworks
- Make testing a habit
- Embed deliberate debugging practice
- Employ scaffolded coding and fade support
- Promote collaboration and explanation

Practical Strategies for Teaching Programming (Part 2)

Developing structured, efficient and decomposed solutions:

- Use Parsons problems regularly
- Focus on structure before syntax
- Model iterative improvement
- Teach decomposition through refactoring
- Compare and critique code

C2. Understanding the structure of Paper 2

- Focus on reading, analysing and refining code
- Linked to Python 3 content in the PLS
- Questions test analysis and refinement, not just writing from scratch
- Typical tasks: fix errors, rearrange code, improve readability, select correct lines
- Final question = full program design (covered in June's webinar)
- Apply standard ways of working – clear, consistent structure

Suggested time: 20 minutes

3 A program allows the user to enter a string.

If the string is fewer than three characters long, an error message is displayed and the string is not processed.

Otherwise, the program adds up the ASCII values for each character in the string.

When the total is divisible by four, a four is added to the string.

When the total is divisible by five, a five is added to the string.

When the total is divisible by both four and five, a four is added to the string.

When the total is not divisible by either four or five, a zero is added to the string.

The program displays the total value and the changed string.

The table shows accurate test results for six inputs.

Input	Output
X	String must be three or more characters long
XY	String must be three or more characters long
cup	328 cup4
snow	455 snow5
lan	280 lan4
HAT	221 HAT0
<empty>	No output, but exits program

The lines of code in the program are mixed up. The indentation has been done for you.

PRIMM & Structure Awareness

Students familiar with PRIMM or Use-Modify-Create will already be used to *reading, predicting and reasoning* about given code before editing it.

Parsons & Scaffolding Practice

Regular Parsons-style classroom tasks train students to recognise structure, indentation and sequencing — exactly the skill needed to re-order these mixed lines correctly.

Testing & Debugging Routines Predict-Run-Compare routines help learners trace logic and verify output step-by-step, matching given test data.

```
Q03.py - Notepad
File Edit Format View Help
# -----
# Constants
# -----
# -----
# Global variables
# -----
# -----
# Main program
# -----

    print ("String must be three or more characters long")
    total = total + asciiValue
    # Check if divisible by 4, 5, or neither
    if (total % FOUR == 0):
        inString = inString + str (ZERO)
    if (len (inString) >= THREE):
ZERO = 0
THREE = 3
    else:
inString = ""
        for character in inString:
            inString = inString + str (FIVE)
            total = 0 # Reset each pass
            elif (total % FIVE == 0):
asciiValue = 0

FOUR = 4
    else:
FIVE = 5
while (inString != ""):
    print (total, inString)
    asciiValue = ord(character)

    inString = input ("Enter a string (blank to exit): ")
# Set up the input loop
inString = input ("Enter a string (blank to exit): ")
    inString = inString + str (FOUR)
```

Suggested time: 25 minutes

4 A program is required to calculate the values in a multiplication square.

Here is an example of a multiplication square.

×	1	2	3	4	5
1	1	2	3	4	5
2	2	4	6	8	10
3	3	6	9	12	15
4	4	8	12	16	20
5	5	10	15	20	25

$2 \times 5 = 10$

The number 10, at the intersection of row 2 and column 5, is the result of 2×5

The user enters a number for the sides of the square into the program.

The program has these requirements:

- calculate the values for the multiplication square
- write the values to a new file
- separate values with commas
- do not include the row and column labels.

Here are the contents of the output file, when the user enters the number 5

```
1, 2, 3, 4, 5
2, 4, 6, 8, 10
3, 6, 9, 12, 15
4, 8, 12, 16, 20
5, 10, 15, 20, 25
```

The program is partially complete.

Comment-First Coding
Students trained to read and interpret scaffolded comments can plan each step before coding, mirroring this task exactly.

Scaffolding to Independence
Regular practice
Completing partially written programs helps learners understand structure, flow, and indentation before they tackle blank-page tasks.

Decomposition and Clarity
Breaking the task into sub-goals (calculate, format, write, close file) mirrors the decomposition process students should rehearse in class.

Q04.py - Notepad

File Edit Format View Help

```
# -----
# Constants
# -----
OUT_FILE = "Q04_OUTPUT.TXT"
COMMA = ","
LF = "\n"

# -----
# Global variables
# -----
outString = ""
maxNum = 0
num = 0
row = 0
column = 0

# -----
# Main program
# -----

# =====> Open the output file for writing

maxNum = int (input ("Enter a number: ")) # Get the number
for row in range (1, maxNum + 1):         # Going down the table
    outString = ""

    for column in range (1, maxNum + 1):   # Going across the table
        # =====> Calculate the new value

        # =====> Add a comma to all except last column

        # =====> Add a line feed to the last column

    # =====> Write the row to the file

# =====> Close the file
```

PLS Quizzes

How about a pop quiz to get students used to using the PLS? It helps reinforce key terminology, familiarises them with what's covered in the PLS and gets them used to the <notation> for arguments. E.g.

Q1: Give four examples of relational operators. (4)

Q2: Which kind of loop is used for iteration? (1)

Q3: What is the (official) name of the % operator in Python? (1)


Q4: What is the difference between a procedure and a function? (1)


Q5: What two parameters are needed to open a file for appending? (2)


Q6: What does ord(<char>) do? (1)


Assessment and feedback in both Years 10 and 11


D1a. Y10


 **Q3_Y10-01-CT6_Assessment**
Due Wed 26 March 2025 at 10:24


 **Q2_Y10-01-CT6_Assessment**
Due Wed 26 March 2025 at 10:23

 **Q1_Y10-01-CT6_Assessment**
Due Wed 26 March 2025 at 10:22

 **Q4_Y10-01-CT6_Assessment**
Due Wed 26 March 2025 at 10:22

 **Q5_Y10-01-CT6_Assessment**
Due Wed 26 March 2025 at 10:21


 **Q6_Y10-01-CT6_Assessment**
Due Wed 26 March 2025 at 10:21

 **Q6_Y10-01-CT6_Assessment** ✓ Submit ▶ Run

1

Assignment

Output

 **Q6_Y10-01-CT6_Assessment**
Due Wed 26 March 2025 at 10:21

Question 6

Write a program that asks the user to enter the amount of money a bill comes to. The program should then calculate the VAT to add on (20%) and output both the amount of VAT and the new total with VAT added on. Variables should have appropriate names and output messages should be useful. (10)

D1b. Y10

Practice Question 1

```
1 # -----
2 # Import libraries
3 # -----
4 # =====> Import the required libraries
5 import turtle
6 # -----
7 # Constants
8 # -----
9 WIDTH = 800
10 HEIGHT = 600
11 SIDE = 90          # length of a square's side
12 GAP = 10           # gap between squares in the grid
13 # -----
14 # Global variables
15 # -----
16 myColours = ["red", "orange", "gold", "olive", "green", "cyan", "navy", "purple", "black"]
17
18 xPos = -145        # lower-left corner of square 1
19 yPos = -135
20
21 # -----
22 # Main program
23 # -----
24 # Window and canvas
25 screen = turtle.Screen()
26 screen.setup(WIDTH, HEIGHT)
27 # -----
28 # =====> Prepare the turtle (speed, pen up/down as needed)
29 # =====> Draw border lines A, B, C, D (black, width 3) to form a rectangle
30 # =====> Draw a 3x3 grid of filled squares inside the border
31 #         - start from (xPos, yPos) which is the lower-left corner of square 1
32 #         - each square has side length SIDE
33 #         - horizontal/vertical spacing between squares is GAP
34 #         - outline pensize is a random integer between 4 and 8 (inclusive)
35 #         - fill colours come from myColours in order, repeating if needed
36 #         - use begin_fill() and end_fill()
37 # -----
```

✓ Submit

▶ Run

Assignment

Output



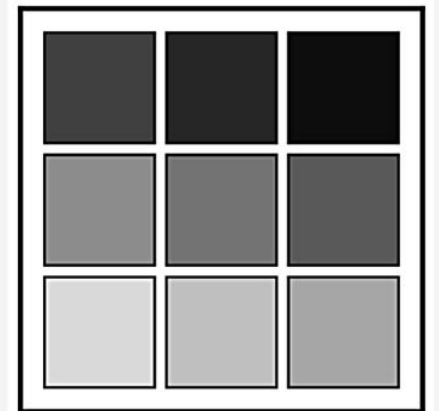
Practice Question 1

Due Fri 24 October 2025 at 12:16

Grid of Squares (Suggested time: 25 minutes)

A program is being developed to display a **3×3 grid of coloured squares** inside a black rectangular border using turtle graphics.

Here is a **greyscale image** of the output produced by a fully functional program.



D2a. Y10 – Pearson assessments 10.1–10.6

‘Real’ assessments:

- Take place in computer room using same format as actual Paper 2 exam.
- Students gain confidence and competence for the real exam.

Mark scheme:

- Colour-code mark requirements to improve understanding of how marks are gained.
- Add in student code and final code as an image for individuals to do corrections.

Question 3

A program is required to draw a pattern on the turtle graphics window.

The program must meet these requirements:

- draw a zigzag pattern
- use only the pen, not shapes
- use every colour in the data structure named ‘colourTable’, in order
- use the <turtle>.setheading(<degrees>) function to control the direction of the zigzag
- use a ‘for in range’ to reduce the need for repeated lines of code.

The output of the program is shown here.



Open the file ‘CT36_Q03_Student.py’.

Amend the code to create a program that meets the requirements above.

Do not add any additional functionality.

Save your amended code as ‘CT36_Q03_Finished.py’.

Copy and paste your code here. (10)

One mark for each up to a maximum of (10).

- Creation of a turtle with a meaningful name. (1)
- Selection of a large pen size. (1)
- Initial position chosen to allow full image to be seen. (1)
- Use of ‘for in range’ loop. (1)
- Every colour from colourTable used. (1)
- Length of leg is appropriate to show design. (1)
- Set heading used to control the zig or zag. (1)
- Heading readjusted each time through the loop. (1)
- Heading readjusted using mathematical calculation rather than ‘if’ statement. (1)
- Output matches image in question. (1)

7 marks

D2b. Y10 diagnostic questions for formative assessment

MCQ01: For loops and the range function

What is the output from this program?

For loops and the range function MCQ01.py ✕

```
1 x = 2
2
3 for i in range (1,4):
4     x = x * i
5
6 print(x)
7
```

A. 5 B. 8 C. 12 D. 48

Starter for lesson about **for loops and in range**.
Three slides. For each slide:

- each student writes answer on a Mini Whiteboard choosing from A, B, C, D
- then they fill in a solo response on google/MS form
- paired peer discussion
- fill in peer response
- teacher shows shared results to all
- class discussion to explore all answers.

Diagnosticquestions.com
[Raspberry Pi Foundation](https://RaspberryPiFoundation.org)
[Quick Read Peer Instruction](#)

D2a. Y10 – Pearson assessments 10.1–10.6

For each paper, explain what you think has happened.

What advice would you give to the Y11 student?

(source: ResultsPlus)

Student B - Paper 2 (38/75)

Question #	Score	Performance	Education - ALE	Variance	Self-assessed
Q01.1	5/5	<div></div>	0.8711	+17.00 %	ⓘ
Q01.2	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q01.3	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q01.4	5/5	<div></div>	0.9911	+17.00 %	ⓘ
Q01.5	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q01.6	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q01.7	5/5	<div></div>	0.8111	+11.00 %	ⓘ
Q01.8	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q01.9	5/5	<div></div>	0.2811	+28.00 %	ⓘ
Q02.1	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q02.10	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q02.2	5/5	<div></div>	0.9011	+17.00 %	ⓘ
Q02.3	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q02.4	5/5	<div></div>	0.7111	+7.00 %	ⓘ
Q02.5	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q02.6	5/5	<div></div>	0.2711	+27.00 %	ⓘ
Q02.7	5/5	<div></div>	0.8111	+11.00 %	ⓘ
Q02.8	5/5	<div></div>	0.7011	+7.00 %	ⓘ
Q02.9	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q03.1	5/5	<div></div>	0.7811	+13.00 %	ⓘ
Q03.10	5/5	<div></div>	0.8511	+15.00 %	ⓘ
Q03.11	5/5	<div></div>	0.8511	+15.00 %	ⓘ
Q03.12	5/5	<div></div>	0.5411	+4.00 %	ⓘ
Q03.13	5/5	<div></div>	0.4811	+4.00 %	ⓘ
Q03.14	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q03.15	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q03.2	5/5	<div></div>	0.7011	+7.00 %	ⓘ
Q03.3	5/5	<div></div>	0.7811	+13.00 %	ⓘ
Q03.4	5/5	<div></div>	0.7711	+12.00 %	ⓘ
Q03.5	5/5	<div></div>	0.4711	+4.00 %	ⓘ
Q03.6	5/5	<div></div>	0.4111	+4.00 %	ⓘ
Q03.7	5/5	<div></div>	0.8311	+13.00 %	ⓘ
Q03.8	5/5	<div></div>	0.5511	+5.00 %	ⓘ
Q03.9	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q04.1	5/5	<div></div>	0.8111	+11.00 %	ⓘ
Q04.2	5/5	<div></div>	0.4811	+4.00 %	ⓘ
Q04.3	5/5	<div></div>	0.2411	+2.00 %	ⓘ
Q04.4	5/5	<div></div>	0.3111	+3.00 %	ⓘ
Q04.5	5/5	<div></div>	0.2811	+2.00 %	ⓘ
Q04.6	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.7	5/5	<div></div>	0.8111	+11.00 %	ⓘ
Q04.8	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q049	5/5	<div></div>	0.5015	+1.00 %	ⓘ
Q05.1	5/5	<div></div>	0.5011	+4.00 %	ⓘ
Q05.2	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q05.3	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q05.4	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q05.5	5/5	<div></div>	0.3911	+3.00 %	ⓘ
Q05.6	5/5	<div></div>	0.2811	+2.00 %	ⓘ
Q05.7	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q05.8	5/5	<div></div>	0.3111	+3.00 %	ⓘ
Q05.9	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q059	5/5	<div></div>	1.4813	+13.33 %	ⓘ
Q0510	5/5	<div></div>	1.0113	+1.00 %	ⓘ
Q06.1	5/5	<div></div>	0.3911	+3.00 %	ⓘ
Q06.2	5/5	<div></div>	0.2811	+2.00 %	ⓘ
Q06.3	5/5	<div></div>	0.1811	+1.00 %	ⓘ
Q06.4	5/5	<div></div>	0.3111	+3.00 %	ⓘ
Q06.5	5/5	<div></div>	0.1611	+1.00 %	ⓘ
Q06.6	5/5	<div></div>	0.3311	+3.00 %	ⓘ
Q069	5/5	<div></div>	0.4813	+4.33 %	ⓘ
Q0699	5/5	<div></div>	0.6013	+6.00 %	ⓘ
Q0610	5/5	<div></div>	0.6813	+6.80 %	ⓘ

Student C - Paper 2 (65/75)

Question #	Score	Performance	Education - ALE	Variance	Self-assessed
Q01.1	5/5	<div></div>	0.8711	+17.00 %	ⓘ
Q01.2	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q01.3	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q01.4	5/5	<div></div>	0.9011	+17.00 %	ⓘ
Q01.5	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q01.6	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q01.7	5/5	<div></div>	0.8311	+13.00 %	ⓘ
Q01.8	5/5	<div></div>	0.8011	+13.00 %	ⓘ
Q01.9	5/5	<div></div>	0.2311	+23.00 %	ⓘ
Q02.1	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q02.10	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q02.2	5/5	<div></div>	0.9011	+17.00 %	ⓘ
Q02.3	5/5	<div></div>	0.8311	+17.00 %	ⓘ
Q02.4	5/5	<div></div>	0.5111	+11.00 %	ⓘ
Q02.5	5/5	<div></div>	0.8011	+12.00 %	ⓘ
Q02.6	5/5	<div></div>	0.7111	+11.00 %	ⓘ
Q02.7	5/5	<div></div>	0.8811	+13.00 %	ⓘ
Q02.8	5/5	<div></div>	0.7711	+12.00 %	ⓘ
Q02.9	5/5	<div></div>	0.8211	+13.00 %	ⓘ
Q03.1	5/5	<div></div>	0.7811	+13.00 %	ⓘ
Q03.10	5/5	<div></div>	0.8511	+15.00 %	ⓘ
Q03.11	5/5	<div></div>	0.8511	+15.00 %	ⓘ
Q03.12	5/5	<div></div>	0.5411	+4.00 %	ⓘ
Q03.13	5/5	<div></div>	0.4811	+4.00 %	ⓘ
Q03.14	5/5	<div></div>	0.4711	+4.00 %	ⓘ
Q03.15	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q03.2	5/5	<div></div>	0.7011	+7.00 %	ⓘ
Q03.3	5/5	<div></div>	0.7711	+12.00 %	ⓘ
Q03.4	5/5	<div></div>	0.7811	+13.00 %	ⓘ
Q03.5	5/5	<div></div>	0.7711	+12.00 %	ⓘ
Q03.6	5/5	<div></div>	0.4711	+4.00 %	ⓘ
Q03.7	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q03.8	5/5	<div></div>	0.5511	+5.00 %	ⓘ
Q03.9	5/5	<div></div>	0.8811	+14.00 %	ⓘ
Q04.1	5/5	<div></div>	0.8311	+13.00 %	ⓘ
Q04.2	5/5	<div></div>	0.4811	+4.00 %	ⓘ
Q04.3	5/5	<div></div>	0.2411	+2.00 %	ⓘ
Q04.4	5/5	<div></div>	0.3111	+3.00 %	ⓘ
Q04.5	5/5	<div></div>	0.2811	+2.00 %	ⓘ
Q04.6	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.7	5/5	<div></div>	0.8111	+11.00 %	ⓘ
Q04.8	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.9	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.10	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.11	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.12	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.13	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.14	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.15	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.16	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.17	5/5	<div></div>	0.5011	+5.00 %	ⓘ
Q04.18	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.19	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.20	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.21	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.22	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.23	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.24	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.25	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.26	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.27	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.28	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.29	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.30	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.31	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.32	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.33	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.34	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.35	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.36	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.37	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.38	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.39	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.40	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.41	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.42	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.43	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.44	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.45	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.46	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.47	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.48	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.49	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.50	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.51	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.52	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.53	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.54	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.55	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.56	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.57	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.58	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.59	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.60	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.61	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.62	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.63	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.64	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.65	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.66	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.67	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.68	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.69	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.70	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.71	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.72	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.73	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.74	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.75	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.76	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.77	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.78	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.79	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.80	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.81	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.82	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.83	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.84	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.85	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.86	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.87	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.88	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.89	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.90	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.91	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.92	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.93	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.94	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.95	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.96	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.97	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.98	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q04.99	5/5	<div></div>	0.5111	+5.00 %	ⓘ
Q0500	5/5	<div></div>	0.5111	+5.00 %	ⓘ

Student D - Paper 2 (75/75)

Question #	Score	Performance %	Education - Avg	Variance %	Self-assessed
Q01.1	5/5	<div></div>	0.8711	+17.00 %	ⓘ
Q01.2	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q01.3	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q01.4	5/5	<div></div>	0.9011	+17.00 %	ⓘ
Q01.5	5/5	<div></div>	0.9011	+14.00 %	ⓘ
Q01.6	5/5	<div></div>	0.9011	+16.00 %	ⓘ
Q01.7	5/5	<div></div>	0.8911	+19.00 %	ⓘ
Q01.8	5/5	<div></div>	0.9211	+14.00 %	ⓘ
Q01.9	5/5	<div></div>	0.9311	+17.00 %	ⓘ
Q01.10	5/5	<div></div>	0.8911	+14.00 %	ⓘ
Q02.10	5/5	<div></div>	0.8611	+16.00 %	ⓘ
Q02.2	5/5	<div></div>	0.9011	+17.00 %	ⓘ
Q02.3	5/5	<div></div>	0.8911	+17.00 %	ⓘ
Q02.4	5/5	<div></div>	0.9111	+16.00 %	ⓘ
Q02.5	5/5	<div></div>	0.8811	+19.00 %	ⓘ
Q02.6	5/5	<div></div>	0.7711	+29.00 %	ⓘ
Q02.7	5/5	<div></div>	0.8111	+19.00 %	ⓘ
Q02.8	5/5	<div></div>	0.7511	+25.00 %	ⓘ
Q02.9	5/5	<div></div>	0.8211	+18.00 %	ⓘ
Q03.1	5/5	<div></div>	0.7611	+26.00 %	ⓘ
Q03.10	5/5	<div></div>	0.8511	+19.00 %	ⓘ
Q03.11	5/5	<div></div>	0.8511	+19.00 %	ⓘ
Q03.12	5/5	<div></div>	0.5411	+46.00 %	ⓘ
Q03.13	5/5	<div></div>	0.4011	+54.00 %	ⓘ
Q03.14	5/5	<div></div>	0.6111	+40.00 %	ⓘ
Q03.15	5/5	<div></div>	0.3111	+70.00 %	ⓘ
Q03.2	5/5	<div></div>	0.7511	+25.00 %	ⓘ
Q03.3	5/5	<div></div>	0.7611	+26.00 %	ⓘ
Q03.4	5/5	<div></div>	0.7711	+25.00 %	ⓘ
Q03.5	5/5	<div></div>	0.4211	+57.00 %	ⓘ
Q03.6	5/5	<div></div>	0.4011	+56.00 %	ⓘ
Q03.7	5/5	<div></div>	0.6211	+47.00 %	ⓘ
Q03.8	5/5	<div></div>	0.5511	+46.00 %	ⓘ
Q03.9	5/5	<div></div>	0.6611	+44.00 %	ⓘ
Q04.1	5/5	<div></div>	0.8111	+39.00 %	ⓘ
Q04.2	5/5	<div></div>	0.4811	+52.00 %	ⓘ
Q04.3	5/5	<div></div>	0.3411	+60.00 %	ⓘ
Q04.4	5/5	<div></div>	0.3111	+70.00 %	ⓘ
Q04.5	5/5	<div></div>	0.2411	+76.00 %	ⓘ
Q04.6	5/5	<div></div>	0.5311	+60.00 %	ⓘ
Q04.7	5/5	<div></div>	0.6111	+50.00 %	ⓘ
Q04.8	5/5	<div></div>	0.5111	+56.00 %	ⓘ
Q04.9	3/3	<div></div>	0.9311	+49.33 %	ⓘ
Q05.1	5/5	<div></div>	0.5411	+46.00 %	ⓘ
Q05.2	5/5	<div></div>	0.6311	+39.00 %	ⓘ
Q05.3	5/5	<div></div>	0.6911	+31.00 %	ⓘ
Q05.4	5/5	<div></div>	0.6611	+34.00 %	ⓘ
Q05.5	5/5	<div></div>	0.3911	+61.00 %	ⓘ
Q05.6	5/5	<div></div>	0.3911	+71.00 %	ⓘ
Q05.7	5/5	<div></div>	0.4511	+55.00 %	ⓘ
Q05.8	5/5	<div></div>	0.9111	+40.00 %	ⓘ
Q05.9	5/5	<div></div>	0.8511	+45.00 %	ⓘ
Q05.9A	3/3	<div></div>	1.4911	+56.33 %	ⓘ
Q05.9B	3/3	<div></div>	1.0111	+66.33 %	ⓘ
Q06.1	5/5	<div></div>	0.3911	+61.00 %	ⓘ
Q06.2	5/5	<div></div>	0.2911	+71.00 %	ⓘ
Q06.3	5/5	<div></div>	0.1811	+82.00 %	ⓘ
Q06.4	5/5	<div></div>	0.9111	+70.00 %	ⓘ
Q06.5	5/5	<div></div>	0.1611	+84.00 %	ⓘ
Q06.6	5/5	<div></div>	0.3311	+87.00 %	ⓘ
Q06.6A	3/3	<div></div>	0.6611	+84.67 %	ⓘ
Q06.6B	3/3	<div></div>	0.5211	+76.33 %	ⓘ
Q06.6C	3/3	<div></div>	0.6811	+77.33 %	ⓘ

Summary and support

Takeaways

You will:

- be shown options for balancing the theory and programming elements of the course
- see how to organise the existing Pearson CT (Computational Thinking) units with free online platforms and resources
- be shown practical approaches for T&L of programming
- consider forms of assessment and feedback in both Years 10 and 11

Your takeaways:

1. *I will add columns for P1 and P2 to my 2025 SEF to see if there is a residual for theory or programming...*
2. *I am going to do more work on specific Pearson's problem sections of code in response to the poor performance on Qu3 in 2025.*
3. ?
4. ?
5. ?

Future webinar details

Pearson Edexcel GCSE Computer Science Professional Development training sessions

- [Preparing Students for Challenging Concepts and Principles in Paper 1](#)
21 January 2026 at 16:00–17:00 GMT
- [Preparing Students for Challenging Concepts and Principles in Paper 2](#)
04 March 2026 at 16:00–17:00 GMT
- [Tackling 6–Mark Questions in Paper 1](#)
22 April 2026 at 16:00–17:00 BST
- [Deep Dive into Paper 2 – Questions 5 and 6](#)
10 June 2026 at 16:00–17:30 BST

Subject Advisor Support

Computer Science and ICT

Tim Brady



Our subject advisors are experts in their fields and are here to support you throughout the year.

Email: TeachingComputerScience@pearson.com

Phone: +44 (0) 344 463 2535 (Mon–Fri, 9:00–17:00)

[Book an appointment with your Subject Advisor](#)

[Sign up](#) to receive regular updates from your Subject Advisor on qualification news and support for your subject.

Follow our [Facebook page](#) and join the Pearson Edexcel Computer Science [teacher group](#) for private, professional discussions managed by your Subject Advisor.



Qualification Services

The following services are included as part of your qualification fees:



examWizard

An online resource bank of past paper questions and support materials, enabling you to create customised mock exams and tests tailored to your students' needs.



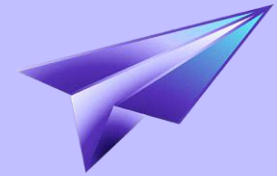
ResultsPlus

An online results analysis tool that provides a detailed breakdown of students' performance in our exams, helping you identify areas for improvement and track progress effectively.



Access To Scripts

An online service that allows you to view and download your candidates' marked scripts. This tool offers transparency and insight into the marking process.



Mocks Service

We offer a paid service for mocks, including paper-based and onscreen assessments. Marked by examiners, results are uploaded to ResultsPlus for detailed analysis to aid student progression and teacher training. Note that this service is not available for all subjects; please check availability for your specific subject.

Find out more

For more professional development courses please see [Pearson's Professional Development Academy](#).



Following this event, you will receive a survey to share your thoughts about the session. Please let us know what you'd like to see more of and what areas could be improved.



