

# EDEXCEL INTERNATIONAL GCSE (9-1) **Computer Science**

## GETTING READY TO TEACH

Event code: 17IBAN01

---

First teaching in 2017, first assessment in 2019.

---



# Aims and Objectives

To equip you with the information you need to successfully plan and deliver international GCSE Computer Science, including:

- XX** the structure and content of the qualification
- XX** how it is assessed, including requirements for the practical examination
- XX** support available from Pearson and others
- XX** teaching and delivery strategies

And to give you an opportunity to share ideas and pool your expertise



# Agenda

10:00 Welcome and introductions

10:10 Overview of the qualification

10:30 Topics 1 & 2

11:05 Paper 02 – Application of  
Computational Thinking

11:45 Topics 3 – 6

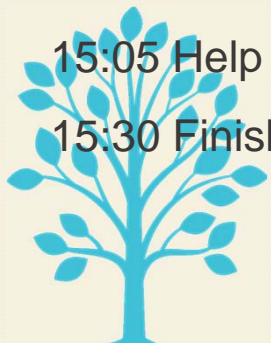
12:30 Lunch

13:30 Paper 01 – Principles of Computer  
Science

14:05 Planning and Delivery

15:05 Help and Support from Pearson

15:30 Finish



# What is computer science?

- Students who study computer science learn how specific computers and applications work and how to fix them when they break.
- Although computer science is an academic discipline with a well-defined body of knowledge, it also has a significant practical problem solving element.
- Computer science is primarily concerned with learning to program.
- Being good at maths is a prerequisite for studying computer science.
- Computer science is just a new name for ICT



# Which of these will students enjoy most?

- Coding a solution to a problem
- Logical thinking challenges
- Learning how a computer works
- Discussing ethical issues arising from the use of computing technology
- All of the above





# Key Features

- Engaging, contemporary content
- Focus on computational thinking
- Examination-only assessment
- Clear and straight-forward question papers
- Choice of three programming languages
- Fosters progression



# 9-1 grading scale (1)

## Awarding

- The grading system is changing but our commitment to awarding grades that accurately reflect learner exam performance remains the same.
- We set new grade boundaries (minimum number of marks needed to achieve each grade) for each assessment of each qualification.

## Benefits

- Greater differentiation across levels of attainment e.g. 2 grades where the current C grade is
- Rewards truly outstanding achievement with the grade 9
- Provides more information about student attainment to help progression to A Level
- Same scale for Pearson Edexcel GCSE and International GCSE allows clear comparison with English standards, unlike old A\* to G grading



# 9-1 grading scale (2)

	NEW 9-1 GRADES	CURRENT A*-G GRADES
The new Grade 9 represents a new level of attainment and we've introduced this to really differentiate your top performing students.	9	A*
	8	
The bottom of the <b>grade 7</b> aligns with the bottom of the grade A.	7	A
	6	B
There's also greater differentiation in the middle range of grades, with <b>grades 4 to 5</b> being equivalent to the old grade B and grade C.	5	
So <b>grade 5</b> will be awarded to the top grade C performers and <b>grade 6</b> to the grade B performers.	4	C
	3	D
The bottom of the <b>grade 4</b> aligns with the bottom of the grade C.	2	E
	1	F
		G
The bottom of the <b>grade 1</b> aligns with the bottom of the grade G.	U	U





# **The Edexcel International GCSE (9-1) Computer Science qualification**



## Overview of the qualification

### Content

1. Problem Solving
2. Programming
3. Data
4. Computers
5. Communication and the internet
6. The bigger picture

### Assessment Objectives

- AO1.** Demonstrate knowledge and understanding of key principles of computer science
- AO2.** Apply knowledge and understanding of key concepts and principles of computer science
- AO3.** Analyse problems in computational terms:
- to make reasoned judgements
  - to design, program, evaluate and refine solutions

### Structure of Assessment

100% external assessment

Fully linear

9 – 1 grading scale

Two papers each with a weighting of 50%

Written examination

Principles of Computer Science

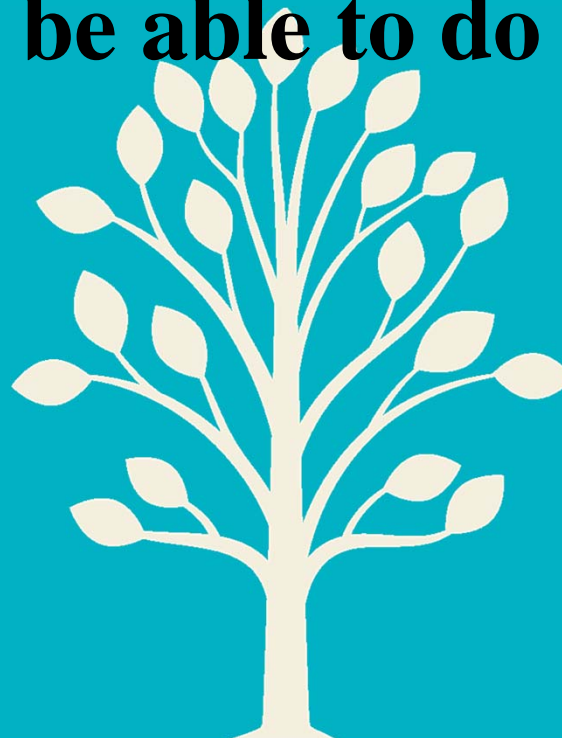
Practical examination

Application of Computational Thinking



# **Problem solving and programming:**

**What students must understand and  
be able to do**



# Problem solving and programming

```

FUNCTION calc_averages(data)
# calculates average temp and wind speed for selected data
BEGIN FUNCTION
SET totalTemp TO 0
SET totalWindSpeed TO 0
SET numReadings TO LENGTH(data)
FOR EACH item FROM data
SET totalTemp TO totalTemp + data[3]
SET totalWindSpeed TO totalWindSpeed + data[4]
ENDFOR
SET averageTemp TO totalTemp/numReadings
SET averageWindSpeed TO totalWindSpeed/numReadings
RETURN averageTemp, averageWindSpeed
END FUNCTION

```

```

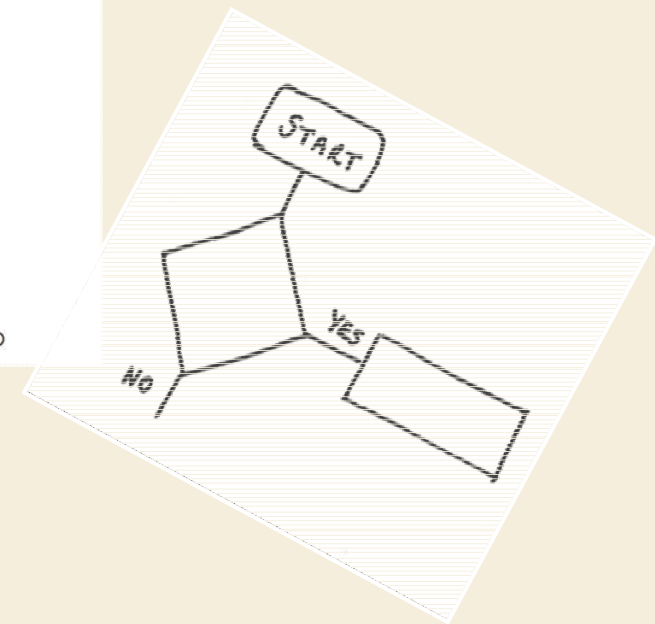
theArtists = [
["Andy", "Warhol", 1928],
["Pablo", "Picasso", 1881],
["Salvador", "Dali", 1904],
["Lavinia", "Fontana", 1552],
["Jackson", "Pollock", 1912],
["Henri", "Matisse", 1869],
["Frida", "Kahlo", 1907],
["Georgia", "O'Keeffe", 1887],
["Kara", "Walker", 1969],
["Yayoi", "Kusama", 1929]
]

theLabels = [] # Put the new user labels into this structure

# Make the artist labels
for person in theArtists:
    newRecord = person[1][0] + person[0][0] + str(person[2])
    theLabels.append(newRecord)
print ("The new userIDs are: ", theLabels)

# Find and print the youngest person and their birthdate
maxDate = 0
for person in theArtists:
    if person[2] > maxDate:
        maxDate = person[2]
        maxPerson = person
print (maxPerson[0], maxPerson[1], "is youngest", str(maxPerson[2]))

```



Write your name here

Surname	Other names
---------	-------------

**Pearson Edexcel**  
**Level 1/Level 2**  
**International GCSE (9–1)**

Centre Number	Candidate Number
---------------	------------------

# Computer Science

## Paper 2: Application of Computational Thinking

Sample assessment material for first teaching September 2017 <b>Time: 3 hours</b>	Paper Reference <b>4CP0/02</b>
---	-----------------------------------

<b>You must have:</b> A computer workstation with appropriate programming language code editing software and tools, including a code interpreter/compiler, CODES folder containing code and data files, pseudocode reference	Total Marks
--	-------------

### Instructions

- Use **black** ink or ball-point pen.
- **Fill in the boxes** at the top of this page with your name, centre number and candidate number.
- Answer **all** questions.
- Answer the questions **requiring a written answer** in the spaces provided – *there may be more space than you need.*
- Only **one** programming language must be used throughout the test.
- Carry out practical tasks on the computer system and save new or amended code using the name given with the appropriate file extension.
- Do **not** overwrite the original code and data files provided to you.
- You must **not** use the Internet during the test.

### Information

- The total mark for this paper is 80.
- The marks for **each** question are shown in brackets
- This paper covers Python, C# and Java.
- The CODES folder in your user area includes all the code and data files you need.
- The Invigilator will tell you where to store your work.

### Advice

- Read each question carefully before you start to answer it.
- Save your work regularly.
- Check your answers and work if you have time at the end.

Turn over ►

S54964A

©2017 Pearson Education Ltd.

1/1/1



S 5 4 9 6 4 A 0 1 1 2



Pearson

43

Pearson Edexcel International GCSE in Information and Communication Technology  
 Sample Assessment Materials – Issue 1 – October 2016 © Pearson Education Limited 2016

edexcel

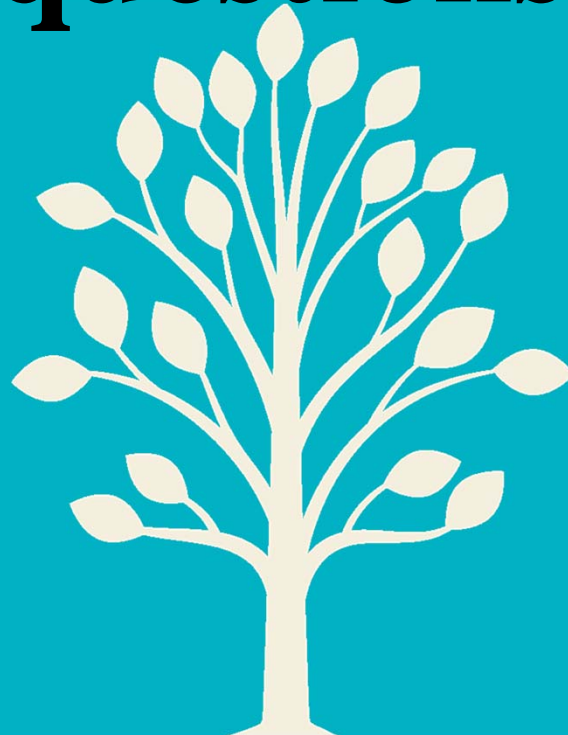
- 3 hour, untiered practical exam
- 5-day window
- Candidates must have:
  - a computer workstation with one of the approved programming language (Python, C# or Java),
  - editing software and tools, including a code translator
  - access to a CODES folder containing the supplied code and data files
  - a pseudocode reference sheet
- Some questions require a written response
- No Internet access
- No choice of questions.
- File naming conventions must be observed





# Activity 1

## **Command words and questions**



# Activity 1 Task 1

## Command words



# Activity 1 Task 2 Question Q01(c)(ii)

```
myNumbers = [10, 20, 30, 40 ,50, 60, 70, 80, 90, 100]
```

```
total
```

```
for theNumber in myNumbers:
```

```
    total = total + theNumber
```

```
    if (theNumber % 2 == 0):
```

```
        print("Even")
```

```
    else:
```

```
        print("Odd")
```

```
print(total)
```



# Conducting the practical exam

- Set up a separate secure user area with sufficient storage for each candidate
- Download the data files, check for compatibility and copy into each candidates' user area
- Check that the computer equipment and software to be used is suitable
- Ensure that at least one invigilator is able to deal with any technical issues that may arise
- Ensure that during the examination candidates :
  - ✓ can only access the files required for the examination
  - ✓ cannot access the internet or refer to textbooks
  - ✓ cannot save the files they produce in a central, unsecure location or on a portable storage device
  - ✓ cannot view each other's screens
  - ✓ can make use of offline help facilities /software-specific manuals
- Return paper scripts to Pearson and upload digital responses
- Refer to the ICE document for further, detailed instructions



# Activity 2

## **Approaches to teaching problem-solving and programming**





# Free learning to program resources

Codecademy: <https://www.codecademy.com>

Grok Learning: <https://groklearning.com>

Sentdex Youtube channel: <https://www.youtube.com/user/sentdex/videos>

Khan Academy: <https://www.khanacademy.org>

Interactive tutorials: <https://learnpython.org>, <https://learnjavaonline.org>,  
<https://www.learnncs.org>

CodingBat: <http://codingbat.com>

Code.org hour of code: <https://code.org>

Invent your own computer games with Python:

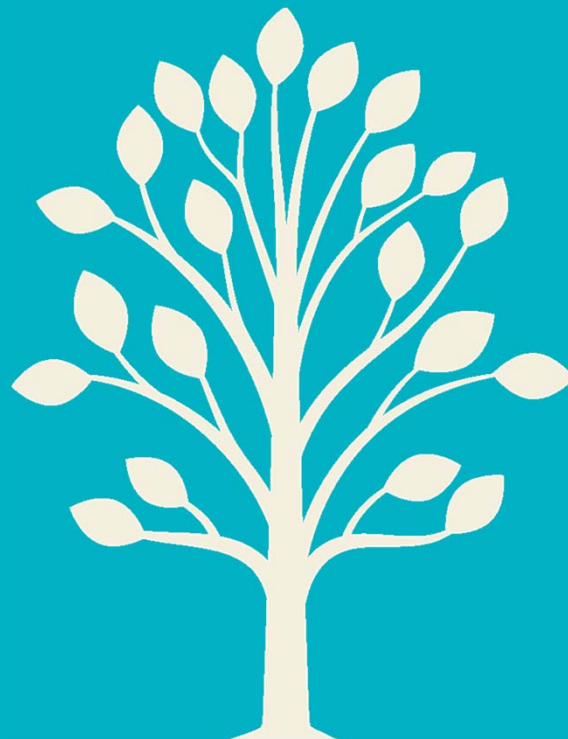
<http://inventwithpython.com/chapters/>

Hacking secret ciphers with Python: <http://inventwithpython.com/hacking/>



# **Introduction to Topics**

## **3 - 6**



# Concepts and principles of Computer Science

## 3. Data

- Binary
- Data representation
- Data storage and compression
- Encryption

## 4. Computers

- Machines and computational models
- Hardware
- Logic
- Software
- Programming languages

## 5. Communication and the internet

- Networks
- Network security
- The internet and the world wide web

## 6. The bigger picture

- Emerging trends, issues and impact



# Computer-related mathematics

Students must be able to:

- ✓ Convert between number bases
- ✓ Perform binary addition, division and multiplication (logical and arithmetic shifts)
- ✓ Construct and interpret expressions and logic statements
- ✓ Convert between units of measurement
- ✓ Calculate file sizes



Write your name here

Surname	Other names
---------	-------------

**Pearson Edexcel**  
**Level 1/Level 2**  
**International GCSE (9–1)**

Centre Number	Candidate Number
---------------	------------------

# Computer Science

## Paper 1: Principles of Computer Science

Sample assessment material for first teaching September 2017 <b>Time: 2 hours</b>	Paper Reference <b>4CP0/01</b>
---	-----------------------------------

<b>You must have:</b> A pseudocode reference	Total Marks
---	-------------

### Instructions

- Use **black** ink or ball-point pen.
- **Fill in the boxes** at the top of this page with your name, centre number and candidate number.
- Answer **all** questions.
- Answer the questions in the spaces provided  
– *there may be more space than you need.*

### Information

- The total mark for this paper is 80.
- The marks for **each** question are shown in brackets  
– *use this as a guide as to how much time to spend on each question.*
- You are not allowed to use a calculator.

### Advice

- Read each question carefully before you start to answer it.
- Try to answer every question.
- Check your answers if you have time at the end.
- Marks will not be awarded for using product or trade names in answers without giving further explanation.

Turn over ►

S54963A

©2017 Pearson Education Ltd.  
1/1



S 5 4 9 6 3 A 0 1 2 2

**Pearson**

5

Pearson Edexcel International GCSE in Information and Communication Technology  
Sample Assessment Materials – Issue 1 – October 2016 © Pearson Education Limited 2016

edexcel

- 2 hour, untiered written exam
- No choice of questions
- Variety of question types, including some multiple choice
- Consistent use of command words
- At least one extended-writing question worth 6 marks
- Some questions involve working with algorithms
- Use of a calculator is not allowed
- Space for drafting where appropriate



**Pearson**

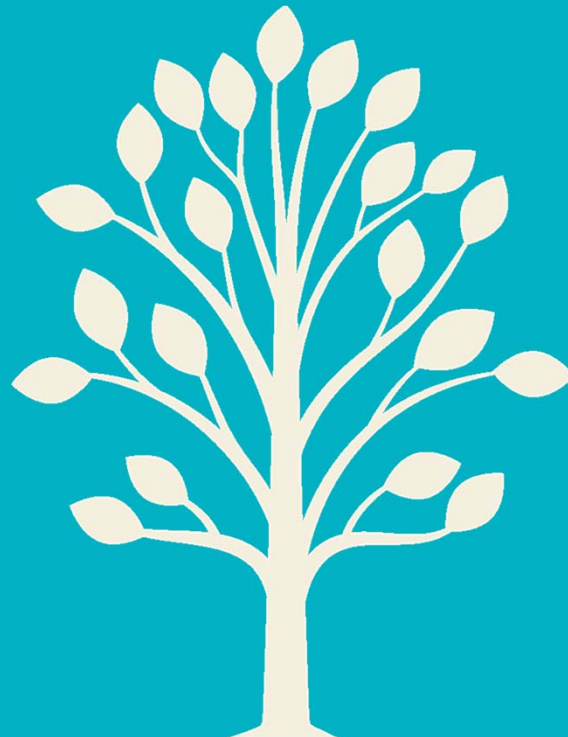


# Activity 3

In your booklet -



# Course Structure



# Assessment structure

Paper 01: Principles of Computer Science

2 hours

50%

Paper 02: Application of Computational Thinking

3 hours

50%



# Subject topics by paper

Topics	Paper 01	Paper 02
1. Problem Solving	✓	✓
2. Programming	✓	✓
3. Data	✓	✓
4. Computers	✓	✓
5. Communication and the internet	✓	✓
6. The Bigger Picture	✓	✓



# Considerations

- Designed to be delivered in 120 – 140 guided learning hours (approximately two 1 hour lessons a week over two years)
- Both the exams must be taken at the end of the course
- Problem solving and programming must be developed and practised throughout the course
- How much experience of programming students already have
- Students won't be using computers in every lesson
- There are opportunities to link theory and practical work



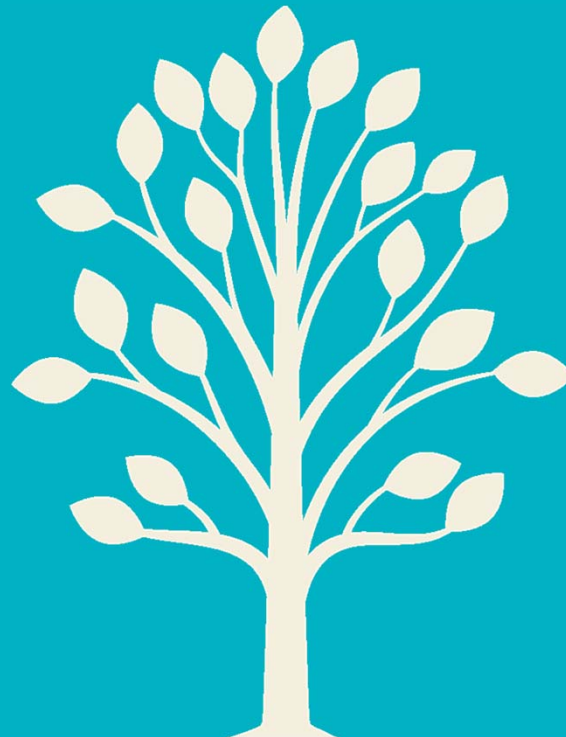


# Activity 4

In your booklet -



# **Planning and Delivery – free support materials**





## Resources

We offer a range of free and paid for resources for International GCSEs. These have been designed to support teachers to improve learner outcomes





## 2-year course planner for Pearson International GCSE Computer Science (9 - 1)

**Weeks** 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

### Year 1, Term 1 (15 weeks)

Lesson 1	Intro	Problem solving & programming	Computers	Representation of numbers	Programming languages	Hardware	OS
Lesson 2	Problem solving & programming						OS

**Weeks** 1 2 3 4 5 6 7 8 9 10 11 12

### Year 1, Term 2 (12 weeks)

Year 1, Term 2 (12 weeks)					
Lesson 1	Networks		Boolean logic		Data rep: text
Lesson 2	Problem solving & programming	The bigger picture	Problem solving & programming	The bigger picture	Networks

**Weeks** 1 2 3 4 5 6 7 8 9 10 11 12 13 14

### Year 1, Term 3 (14 weeks)

Lesson 1	Data rep: images	Data rep: sound	Hardware: internal components	Comput. models	Network security
Lesson 2	Problem solving & programming				Network security

**Weeks** 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

### Year 2, Term 1 (15 weeks)

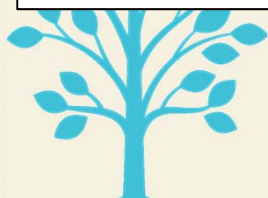
Lesson 1	Prob solv & prog	Data storage and compression	Hardware: secondary storage	Internet & WWW	Encryption	Embedded systems
Lesson 2	Problem solving & programming				Encryption	Big.Pict

**Weeks** 1 2 3 4

### Year 2, Term 2 (4 weeks)

Lesson 1	Bigger picture: Emerging trends
Lesson 2	Problem solving & programming

Topic 1: Problem solving	Topic 4: Computers
Topic 2: Programming	Topic 5: Communication and the internet
Topic 3: Data	Topic 6: The bigger picture



# Scheme of work materials

Week 4 Lesson	Spec ref	Lesson summary	Lesson content	Lesson resources	Transferable skills
1	3.3.2 3.3.3 3.3.4	Data storage and compression: RLE	Show the YouTube video 'Run Length Encoding Visualization' to introduce the concept of RLE. <a href="http://www.youtube.com/watch?v=yqdNscvym_E">http://www.youtube.com/watch?v=yqdNscvym_E</a> Give students the opportunity to experience RLE compression in action by decompressing a 4-bit colour image encoded using RLE. (Activity 4.1.1) Explain how to calculate the size (in bytes) of the uncompressed and the compressed files, pointing out the small size reduction. (Activity 4.1.2) Point out that the advantage of a smaller file is gained at the expense of the computer having to carry out more processing – the file has first to be compressed and then it has to be decompressed before it can be used. This is especially true in this case, where compression only saves one byte. Ask students to consider whether the file would compress more or less if colours were represented by 4-bit codes. Introduce students to the RLE calculator. <a href="http://mathcelebrity.com/runlengthcode.php">http://mathcelebrity.com/runlengthcode.php</a> . Give them an opportunity to try it out and get to grips with how it works. (Activity 4.1.3) Homework: Ask students to complete Activity 4.1.3	Week 4, Lesson 1 activities YouTube video: 'Run Length Encoding Visualization' RLE calculator	Critical thinking Self-direction
2	1.1.6 3.3.3	Data storage and compression: RLE	Ask students to work in pairs to write and test an RLE algorithm they completed for the last homework. Homework: Ask students to complete the summary sheet. (Activity 4.2.2)		

## Activity 4.2.1

Program solution provided in [ProgCode](#) folder.

```
#Activity 4.2.1
def run_length_encoding(data):
    rleCode = ""
    length = len(data)
    if length == 0:
        rleCode += ""
    elif length == 1:
        rleCode += str(length) + data
    else:
        count = 1
        index = 1
        while index < length:
            # Checks if it is the same letter.
            if data[index] == data[index - 1]:
                count += 1
            else:
                rleCode += str(count) + data[index - 1]
                count = 1
            index += 1
        # rleCode += data[index - 1] + str(count)
        rleCode += str(count) + data[index - 1]
    return rleCode

#Main program
yourData = input("Enter the data you want to compress:")
compressed = run_length_encoding(yourData)
print("\n", compressed)
```

## Activity 4.2.2 (homework)

Explain what is meant by lossless compression.

Lossless data compression reduces the size of files in such a way that the original data can be perfectly reconstructed from the compressed data – nothing is lost.

## Activity 4.1.4 (homework)

1. Explain how the RLE compression algorithm works.
2. Here is some data used to represent an image. Each pixel is encoded as a character.  
AADACCEFABAAECFGBDGE  
Explain why a RLE algorithm may not be appropriate for encoding this image.
3. Here is a partly completed RLE algorithm expressed as a written description. Fill in the gaps to complete it.
  1. Start with the first character in the string.
  2. Write down the number 1.
  3. Compare the first character with the next character on the right.
  4. If they are the same, \_\_\_\_\_.
  5. If they are not the same, \_\_\_\_\_.
  6. Move on to the next character on the right.
  7. Go back to step 2 and repeat until you reach the end of the string.
  8. \_\_\_\_\_.

## Week 4, Lesson 2 activities

### Activity 4.2.1

Implement the RLE compression algorithm in Python.

Test out your program using these data strings:

- o AAAABBBBBBBBBBCADDDEEFFFFFFFFF
- o ABCABCABCABCABCABCABCABCBCS



# Support overview

Support for  
all subjects

Getting Started  
Guide &  
Scheme of  
Work

Getting Ready  
to Teach Events

Subject  
interpretation of  
transferable  
skills

Subject Advisor

Results Plus

Regional  
Support  
Manager

Curriculum  
Matched  
Publishing

Exemplar  
Marked  
Responses

Additional SAMs

Exam Wizard

Lesson Plans

Topic booklets

Additional support  
for selected  
subjects





# Free support - descriptions

**Getting Started Guide** *includes mapping of changes, content and assessment guidance, course planner and resource list*

**Editable Scheme of Work** *includes activities to support transferable skills development*

**Exam Wizard** *a free exam preparation tool containing a bank of past Edexcel exam questions*

**Results Plus** *free online service giving instant and detailed analysis of your students' exam and mock performance*

**Regional support manager** *access to a regionally based support manager for any query*

**Subject Advisor** *For any subject related query you have. Sign up to mailing list*

**Exemplar** *Marked student responses to SAMs questions*

**Additional SAMs** *An additional set of Sample Assessment Material available as a secure download*



# Transferable Skills

- Skills frameworks adapted to support design of new Edexcel International GCSEs
- Ensure learners acquire skills needed to access Higher Education and fulfilling careers



## Cognitive skills

Core skills brain uses to think, learn and reason – used to carry out any task.



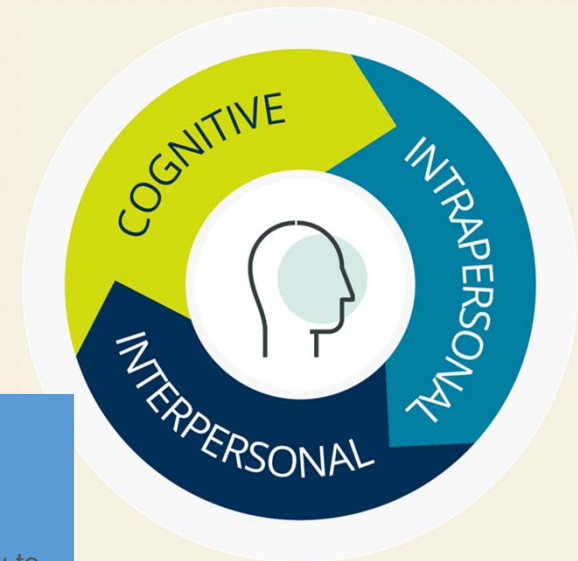
## Intrapersonal Skills

Emotional intelligence, ability to know, understand and manage own emotions and learning.



## Interpersonal Skills

Life skills used every day to communicate and interact with others, individually and in groups.



## Other resources

- Computing as School (CAS)
- CS4FN (Computer Science for Fun)
- Computer Science Unplugged
- Text books for the regulated GCSE in Computer Science
- The Cybersecurity Challenge Schools Programme



# INTERNATIONAL GCSE

## Computer Science (9-1)

### GETTING STARTED GUIDE

Pearson Edexcel International GCSE in Computer Science (4CP0)

For first teaching in September 2017  
First examination June 2019





# World Class Qualifications

- Pearson's World Class Qualification design principles mean all Edexcel qualifications are developed to be **Rigorous**, **Demanding**, **Inclusive** and **Empowering**
- Externally approved by the Expert Panel for World Class Qualifications



# Subject Advisor

**Tim Brady**  
Computer Science and  
ICT Subject Advisor



Phone:  
UK: 020 7010 2161  
Intl: +44 (0) 20 7010 2161

Email:  
[TeachingICT@pearson.com](mailto:TeachingICT@pearson.com)  
[TeachingComputerScience@pearson.com](mailto:TeachingComputerScience@pearson.com)

Twitter:  
[@pearsonICT](https://twitter.com/pearsonICT)  
[@Pearson\\_CS](https://twitter.com/Pearson_CS)

Facebook:  
<https://www.facebook.com/groups/421769201257111/>



# Useful Links

## 1. [Grade Boundaries](#)

This page shows the minimum marks needed to achieve a certain grade for all UK and international examinations. Also refer to the examiners report which is available for download with other documents.

## 2. [Examination Results Statistics](#)

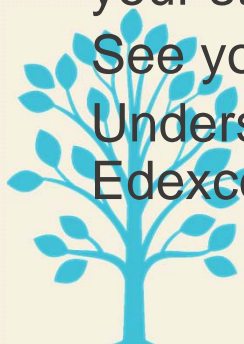
Results statistics summarise the overall grade outcomes of candidates sitting Edexcel examinations.

## 3. [Results Plus](#)

Edexcel's free online service giving instant and detailed analysis of your students' exam and mock performance.

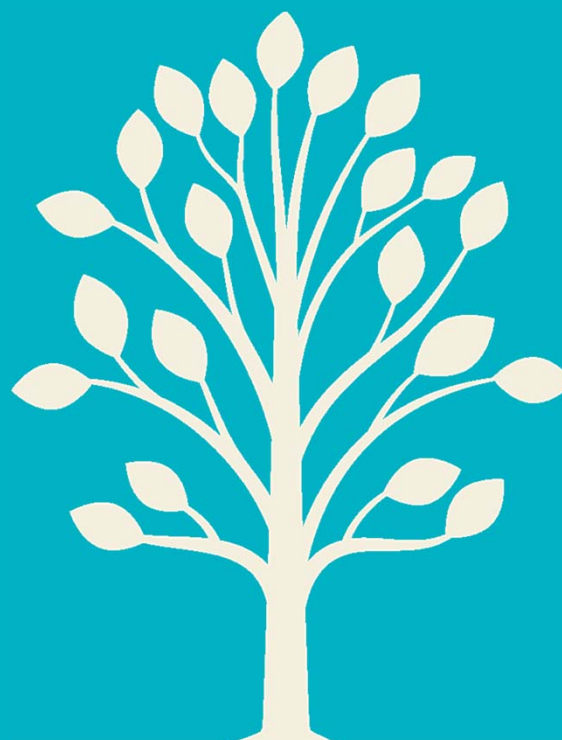
See your students' scores for every exam question.

Understand how your students' performance compares with Edexcel national averages.





# Any further questions?



ALWAYS LEARNING