

INTERNATIONAL GCSE **COMPUTER SCIENCE**

9210/1

Paper 1 Programming

Mark scheme

November 2024

Version: 1.0 Final



2 4 B Y 9 2 1 0 / 1 / M S

Mark schemes are prepared by the Lead Assessment Writer and considered, together with the relevant questions, by a panel of subject teachers. This mark scheme includes any amendments made at the standardisation events which all associates participate in and is the scheme which was used by them in this examination. The standardisation process ensures that the mark scheme covers the students' responses to questions and that every associate understands and applies it in the same correct way. As preparation for standardisation each associate analyses a number of students' scripts. Alternative answers not already covered by the mark scheme are discussed and legislated for. If, after the standardisation process, associates encounter unusual answers which have not been raised they are required to refer these to the Lead Examiner.

It must be stressed that a mark scheme is a working document, in many cases further developed and expanded on the basis of students' reactions to a particular paper. Assumptions about future mark schemes on the basis of one year's document should be avoided; whilst the guiding principles of assessment remain constant, details will change, depending on the content of a particular examination paper.

Further copies of this mark scheme are available from oxfordaqaexams.org.uk

Copyright information

OxfordAQA retains the copyright on all its publications. However, registered schools/colleges for OxfordAQA are permitted to copy material from this booklet for their own internal use, with the following important exception: OxfordAQA cannot give permission to schools/colleges to photocopy any material that is acknowledged to a third party even for internal use within the centre.

Copyright © 2024 OxfordAQA International Examinations and its licensors. All rights reserved.

How to mark

Aims

When you are marking your allocation of scripts your main aims should be to:

- recognise and identify the achievements of students
- where relevant, place students in the appropriate mark band and in the appropriate part of that mark band (high, low, middle) for **each** assessment objective
- record your judgements with brief notes, annotations and comments that are relevant to the mark scheme and make it clear to other associates how you have arrived at the numerical mark awarded for each assessment objective
- ensure comparability of assessment for all students, regardless of question or examiner.

Approach

It is important to be **open-minded** and **positive** when marking scripts.

The specification recognises the variety of experiences and knowledge that students will have. It encourages them to study computer science in a way that is relevant to them. The questions have been designed to give them opportunities to discuss what they have found out about computer science. It is important to assess the quality of **what the student offers**.

Do not mark scripts based on the answer **you** would have written. The mark schemes have been composed to assess **quality of response** and not to identify expected items of knowledge.

Assessment Objectives

This component requires students to:

AO1: Demonstrate knowledge and understanding of the key concepts and principles of computer science.

AO2: Apply knowledge and understanding of key concepts and principles of computer science.

AO3: Analyse problems in computational terms in order to develop and test programmed solutions and demonstrate an understanding of programming concepts.

The following annotation is used in the mark scheme.

; means a single mark

// means alternative response

/ means an alternative word or sub-phrase

A means acceptable creditworthy answer

R means reject answer as not creditworthy

NE means not enough

I means ignore

DPT in some questions a specific error made by a candidate, if repeated, could result in the loss of more than one mark. The **DPT** label indicates that this mistake should result in a candidate losing only one mark, on the first occasion that the error is made.

Provided that the answer remains understandable, subsequent marks should be awarded as if the error was not being repeated.

Section A

Question	Part	Marking guidance	Total marks
01	1	<code>createImage;</code>	1 AO3=1

Question	Part	Marking guidance	Total marks
01	2	<code>row // col;</code>	1 AO3=1

Question	Part	Marking guidance	Total marks
01	3	The joining together of (two or more) strings // combining strings;	1 AO3=1

Question	Part	Marking guidance	Total marks
01	4	When adding 1 to each pixel value; the value may then be larger than 4 // may become 5; the last character / position / index in the <code>GRAPHICS</code> constant / string is 4; a value of 5 is too large / out of bounds (for the <code>GRAPHICS</code> string); MAX 3	3 AO3=3

Question	Part	Marking guidance	Total marks
01	5	The user may enter a <code>q</code> (rather than <code>Q</code>) // the user may enter a lowercase letter; This would cause the error message to be displayed // the program would not exit; So that the user can enter either <code>q</code> or <code>Q</code> to quit the program;; MAX 2	2 AO3=2

Question	Part	Marking guidance	Total marks
02	1	A sequence of instructions; to complete a task // solve a problem;	2 AO3=2

Question	Part	Marking guidance	Total marks
02	2	The list/array uses zero-indexing // the first item in the list/array is at index/position 0; To stop the FOR loop trying to go past the last item in the list/array; MAX 1	1 AO3 = 1

Question	Part	Marking guidance	Total marks
02	3	Change of FOR to REPEAT / WHILE (structure); Creation/Use of a counter variable; Criteria check against counter variable; Counter variable incremented inside iteration / REPEAT / WHILE structure; A. response given as code, mark points mapped to code statements MAX 3	3 AO3 = 3

Question	Part	Marking guidance	Total marks
02	4	3, 2, 1, 4, 0, 3, 2, 1, 4 1 mark all even index positions changed as shown; 1 mark all odd index positions left as original value; I. formatting	2 AO3 = 2

Question	Part	Marking guidance	Total marks
----------	------	------------------	-------------

03	1	<table><tr><th>current</th><th>count</th><th>position</th><th>item</th><th>OUTPUT</th></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td></td></tr><tr><td></td><td>2</td><td>2</td><td>2</td><td>(2,1)</td></tr><tr><td>2</td><td>1</td><td>3</td><td>2</td><td></td></tr><tr><td></td><td>2</td><td>4</td><td>3</td><td>(2,2)</td></tr><tr><td>3</td><td>1</td><td>5</td><td>3</td><td></td></tr><tr><td></td><td>2</td><td>6</td><td>3</td><td></td></tr><tr><td></td><td>3</td><td>7</td><td>3</td><td></td></tr><tr><td></td><td>4</td><td>8</td><td>4</td><td>(4,3)</td></tr><tr><td>4</td><td>1</td><td></td><td></td><td>(1,4)</td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></table>					current	count	position	item	OUTPUT	1	1	1	1			2	2	2	(2,1)	2	1	3	2			2	4	3	(2,2)	3	1	5	3			2	6	3			3	7	3			4	8	4	(4,3)	4	1			(1,4)						6 AO3=6
		current	count	position	item	OUTPUT																																																								
		1	1	1	1																																																									
			2	2	2	(2,1)																																																								
		2	1	3	2																																																									
			2	4	3	(2,2)																																																								
		3	1	5	3																																																									
			2	6	3																																																									
			3	7	3																																																									
			4	8	4	(4,3)																																																								
		4	1			(1,4)																																																								
		1 mark: First value of current is 1 and count is 1 1 mark: First value of position is 1 and item is 1 1 mark: Subsequent row count is 2, position is 2, item is 2 1 mark: First OUTPUT value is (2,1) 1 mark: Correctly traces stages for current being 2 1 mark: Completes rest of trace table correctly I. missing brackets and commas																																																												

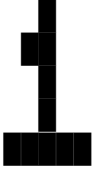
Question	Part	Marking guidance	Total marks
----------	------	------------------	-------------

03	2	To output the last run from the image // the last run will not be output inside the loop; A. for VB only, to return to the calling subroutine	1 AO3 = 1
----	---	--	--------------------------------

Section B

Question	Part	Marking guidance	Total marks
04	1	1 mark: Program has call to <code>displayImage</code> inside the <code>rleImage</code> subroutine. 1 mark: Statement at a position so that the image is displayed before the run length encoded data. 1 mark: Blank line between the image and the run length encoded output. Max 2 marks if any errors Python <pre>def rleImage(width, height, image): displayImage(width, height, image) print() current = image[0] count = 1 for position in range(1, len(image)): item = image[position] if item != current: print(f"({count},{current})",end="") current = item count = 1 else: count += 1 print(f"({count},{current})")</pre> C# <pre>static void rleImage(int width, int height, int[] image) { displayImage(width, height, image); Console.WriteLine(); int current = image[0]; int count = 1; for (int position = 1; position < image.Length; position++) { int item = image[position]; if (item != current) { Console.Write(\$"({count},{current})"); current = item;</pre>	3 AO3=3

		<pre> count = 1; } else { count++; } } Console.WriteLine(\$"{count},{current}"); } VB Sub rleImage(width As Integer, height As Integer, image as Integer()) displayImage(width, height, image) Console.WriteLine() Dim current As Integer = image(0) Dim count As Integer = 1 For position As Integer = 1 To image.Length - 1 Dim item As Integer = image(position) If item <> current Then Console.Write(\$"{count},{current}") current = item count = 1 Else count = count + 1 End If Next Console.WriteLine(\$"{count},{current}") End Sub </pre>	
--	--	--	--

Question	Part	Marking guidance	Total marks
04	2	Screen capture(s) must match code from Q4.1 1 mark: Image is displayed above all other output, with a blank line immediately below the displayed image. <pre> 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: 1 Please enter the image name: one 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: 3  (2,0) (1,4) (3,0) (2,4) (4,0) (1,4) (4,0) (1,4) (2,0) (5,4) 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: </pre>	1 AO3=1

Question	Part	Marking guidance	Total marks
05	1	1 mark: Output statement in appropriate place. 1 mark: Output statement includes use of <code>width</code> variable. 1 mark: Output statement includes use of <code>height</code> variable. 1 mark: Output matches requested text with width and height in the correct order. Python <pre>def main(): image = [] width = 0 height = 0 userChoice = "" while userChoice != "Q": displayMenu() userChoice = input("Please enter a choice: ").upper() match userChoice: case "1": width, height, image = loadImage() print(f"The image has size {width} by {height} pixels") case "2": displayImage(width, height, image) case "3": rleImage(width, height, image) case "4": width, height, image = stretchImage(width, height, image) case "5": image = lightenImage(image) case "6": image = darkenImage(image) case "Q": message = random.randint(0,1) if message == 0: print("Thanks for using this software") else: print("Goodbye") case _: print() print("You did not enter a valid option") print("Press enter to continue") input()</pre>	4 A03=4

		C# <pre> static void Main(string[] args) { int[] image = null; int width = 0; int height = 0; string userChoice = ""; Random message = new Random(); while (userChoice != "Q") { displayMenu(); Console.WriteLine("Please enter a choice: "); userChoice = Console.ReadLine().ToUpper(); switch (userChoice) { case "1": (width, height, image) = loadImage(); Console.WriteLine(\$"The image has size {width} by {height} pixels"); break; case "2": displayImage(width, height, image); break; case "3": rleImage(width, height, image); break; case "4": (width, height, image) = stretchImage(width, height, image); break; case "5": lightenImage(image); break; case "6": darkenImage(image); break; case "Q": if (message.Next(0, 1) == 0) { Console.WriteLine("Thanks for using this software"); } else { </pre>	
--	--	--	--

		<pre> Console.WriteLine("Goodbye"); } break; default: Console.WriteLine(); Console.WriteLine("You did not enter a valid option"); break; } } Console.WriteLine("Press enter to continue"); Console.ReadLine(); } </pre> VB <pre> Sub Main(args() As String) Dim image As Integer() Dim width As Integer = 0 Dim height As Integer = 0 Dim userChoice As String = "" Dim message As Random = New Random While userChoice <> "Q" displayMenu() Console.WriteLine("Please enter a choice: ") userChoice = Console.ReadLine().ToUpper() Select Case userChoice Case "1" Dim values As (width As Integer, height As Integer, image as Integer()) = loadImage() width = values.width height = values.height image = values.image Console.WriteLine(\$"The image has size {width} by {height} pixels") Case "2" displayImage(width, height, image) Case "3" rleImage(width, height, image) Case "4" Dim values As (width As Integer, height As Integer, image as Integer()) = stretchImage(width, height, image) width = values.width height = values.height image = values.image </pre>	
--	--	--	--

		<pre> case "5" lightenImage(image) Case "6" darkenImage(image) Case "Q" If message.Next(0, 1) = 0 Console.WriteLine("Thanks for using this software") Else Console.WriteLine("Goodbye") End If Case Else Console.WriteLine() Console.WriteLine("You did not enter a valid option") End Select End While Console.WriteLine("Press enter to continue") Console.ReadLine() End Sub </pre>	
--	--	---	--

Question	Part	Marking guidance	Total marks
----------	------	------------------	-------------

05	2	Screen capture(s) must match code from Q5.1 1 mark: Appropriate message displayed for the size of the image. <pre> 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: 1 Please enter the image name: pattern The image has size 5 by 9 pixels 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit </pre>	1 AO3=1
-----------	----------	--	-------------------------------------

		Please enter a choice:	
--	--	------------------------	--

Question	Part	Marking guidance	Total marks
06	1	1 mark: Use of indefinite iteration. 1 mark: Test against <code>factor</code> variable. 1 mark: Correct criteria of <code>>5</code> 1 mark: Error message displayed and inside iteration. 1 mark: If input is invalid code will ask again for another input. Max 4 marks if any errors Python <pre>def stretchImage(width, height, image): factor = int(input("Please enter the stretch factor: ")) while factor > 5: print("Invalid input - please try again") factor = int(input("Please enter the scale factor: ")) newWidth = width * factor newHeight = height newImage = createImage(newWidth, newHeight) counter = 0 for position in range(len(image)): for pixelRepeat in range(factor): newImage[counter] = image[position] counter += 1 return newWidth, newHeight, newImage</pre> C# <pre>static (int, int, int[]) stretchImage(int width, int height, int[] image) { Console.Write("Please enter the stretch factor: "); int factor; factor = Convert.ToInt32(Console.ReadLine()); while (factor > 5) { Console.WriteLine("Invalid input - please try again"); factor = Convert.ToInt32(Console.ReadLine()); } int newWidth = width * factor; int newHeight = height; int[] newImage = createImage(newWidth, newHeight); int counter = 0;</pre>	5 AO3=5

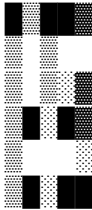
		<pre> for (int position = 0; position < image.Length; position++) { for (int pixelRepeat = 0; pixelRepeat < factor; pixelRepeat++) { newImage[counter] = image[position]; counter++; } } return (newWidth, newHeight, newImage); } </pre> VB <pre> Function stretchImage(width As Integer, height As Integer, image as Integer()) As (Integer, Integer, Integer()) Console.WriteLine("Please enter the stretch factor: ") Dim factor As Integer factor = Convert.ToInt32(Console.ReadLine()) While factor > 5 Console.WriteLine("Invalid input - please try again") factor = Convert.ToInt32(Console.ReadLine()) End While Dim newWidth As Integer = width * factor Dim newHeight As Integer = height Dim newImage As Integer() = createImage(newWidth, newHeight) Dim counter As Integer = 0 For position As Integer = 0 To image.Length - 1 For pixelRepeat As Integer = 0 To factor - 1 newImage(counter) = image(position) counter = counter + 1 Next Next Return (newWidth, newHeight, newImage) End Function </pre>	
--	--	--	--

Question	Part	Marking guidance	Total marks
06	2	Screen capture(s) must match code from Q6.1 1 mark: Shows the values being entered with appropriate error message. <pre> 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: 1 Please enter the image name: one The image has size 5 by 5 pixels 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: 4 Please enter the stretch factor: 6 Invalid input - please try again Please enter the scale factor: 3 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: </pre>	1 A03=1

Question	Part	Marking guidance	Total marks
07	1	1 mark: Selection statement used. 1 mark: Criteria for filename equal to "random". 1 mark: createImage called with width 5 and height 6 1 mark: Iteration used. 1 mark: Iteration runs across appropriate index positions. 1 mark: Random integer picked between 0 and 4 1 mark: Pixel data updated with random integer. 1 mark: Selection statement separates appropriately random image and opening a file // else statement // appropriate return statement. Max 7 marks if any errors Python <pre>def loadImage(): filename = input("Please enter the image name: ") if filename == "random": newImage = createImage(5, 6) for loop in range(0, 30): newImage[loop] = random.randint(0, 4) return 5, 6, newImage else: filename = filename + ".txt" with open(filename) as f: width = int(f.readline()) height = int(f.readline()) newImage = createImage(width, height) for row in range(height): rowData = f.readline() for col in range(width): newImage[row * width + col] = int(rowData[col]) return width, height, newImage</pre> C# <pre>static (int, int, int[]) loadImage() { Console.WriteLine("Please enter the image name: "); string filename = Console.ReadLine(); if (filename == "random") {</pre>	8 AO3=8

		<pre> int[] newImage = createImage(5, 6); Random r = new Random(); for (int a = 0; a < 30; a++) { newImage[a] = r.Next(0, 5); } return (5, 6, newImage); } else { filename = filename + ".txt"; using (StreamReader MyStream = new StreamReader(filename)) { int width = Convert.ToInt32(MyStream.ReadLine()); int height = Convert.ToInt32(MyStream.ReadLine()); int[] newImage = createImage(width, height); for (int row = 0; row < height; row++) { string rowData = MyStream.ReadLine(); for (int col = 0; col < width; col++) { newImage[row * width + col] = Convert.ToInt32(rowData[col].ToString()); } } return (width, height, newImage); } } } </pre> VB <pre> Function loadImage() As (Integer, Integer, Integer()) Console.WriteLine("Please enter the image name: ") Dim filename As String = Console.ReadLine() If filename = "random" Then Dim newImage As Integer() = createImage(5, 6) Dim r as Random = New Random For a As Integer = 0 To 29 </pre>	
--	--	---	--

		<pre> newImage(a) = r.Next(0,5) Next Return (5, 6, newImage) Else filename = filename + ".txt" Using MyStream As StreamReader = New StreamReader(filename) Dim width As Integer = Convert.ToInt32(MyStream.ReadLine()) Dim height As Integer = Convert.ToInt32(MyStream.ReadLine()) Dim newImage As Integer() = createImage(width, height) For row As Integer = 0 To height - 1 Dim rowData As String = MyStream.ReadLine() For col As Integer = 0 To width - 1 newImage(row * width + col) = Convert.ToInt32(rowData(col).ToString()) Next Next Return (width, height, newImage) End Using End If End Function </pre>	
--	--	--	--

Question	Part	Marking guidance	Total marks
07	2	Screen capture(s) must match code from Q7.1 1 mark: Creates random image of size 5 by 6 and displays this image. <pre> 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit </pre> Please enter a choice: 1 Please enter the image name: random The image has size 5 by 6 pixels <pre> 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit </pre> Please enter a choice: 2  <pre> 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit </pre> Please enter a choice:	1 A03=1

Question	Part	Marking guidance	Total marks
08	1	1 mark: Variable defined to store storage space needed for RLE. 1 mark: 2 added to variable inside selection statement. 1 mark: Final 2 added to variable in appropriate place. 1 mark: Calculates correct size of original image. 1 mark: Displays correct message with their calculated original image size. 1 mark: Displays correct message with their calculated RLE image size. ALTERNATIVE METHOD MP2 and MP3: 1 mark: Counts number of runs. 1 mark: Multiplies this by 2. I. minor typos and incorrect case. Max 5 marks if any errors Python <pre>def rleImage(width, height, image): displayImage(image) print() current = image[0] count = 1 rleStorage = 0 for position in range(1, len(image)): item = image[position] if item != current: print(f"({count},{current})",end="") current = item count = 1 rleStorage += 2 else: count += 1 print(f"({count},{current})") rleStorage += 2 print() print(f"The original image needs {width * height} bytes") print(f"The RLE image needs {rleStorage} bytes")</pre> C# <pre>static void rleImage(int width, int height, int[] image) { int current = image[0];</pre>	6 AO3=6

		<pre> int count = 1; int rleStorage = 0; for (int position = 1; position < image.Length; position++) { int item = image[position]; if (item != current) { Console.Write(\$"({count},{current})"); current = item; count = 1; rleStorage += 2; } else { count++; } } rleStorage += 2; Console.WriteLine(\$"({count},{current})"); Console.WriteLine(\$"The original image needs {width * height} bytes"); Console.WriteLine(\$"The RLE image needs {rleStorage} bytes"); } VB Sub rleImage(width As Integer, height As Integer, image as Integer()) Dim current As Integer = image(0) Dim count As Integer = 1 Dim rleStorage As Integer = 0 For position As Integer = 1 To image.Length - 1 Dim item As Integer = image(position) If item <> current Then Console.Write(\$"({count},{current})") current = item count = 1 rleStorage = rleStorage + 2 Else count = count + 1 End If Next Console.WriteLine(\$"({count},{current})") rleStorage = rleStorage + 2 Console.WriteLine() </pre>	
--	--	--	--

		<pre> Console.WriteLine(\$"The original image needs {width * height} bytes") Console.WriteLine(\$"The RLE image needs {rleStorage} bytes") End Sub</pre>	
--	--	---	--

Question	Part	Marking guidance	Total marks
08	2	Screen capture(s) must match code from Q8.1 1 mark: Messages on screen to show original image size of 25 bytes and RLE image size of 14 bytes. <pre> 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: 1 Please enter the image name: zero The image has size 5 by 5 pixels 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: 3 (6,4) (3,0) (2,4) (3,0) (2,4) (3,0) (6,4) The original image needs 25 bytes The RLE image needs 14 bytes 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: </pre>	1 A03=1

Section C

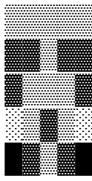
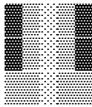
Question	Part	Marking guidance	Total marks
09	1	1 mark: Subroutine with correct parameters <code>width</code> and <code>image</code>. 1 mark: String variable defined and set to an empty string (Python) // declared (not necessarily set to empty string for C# and VB.NET). 1 mark: Iteration running from index 0 to index <code>len(image)</code> 1 mark: Iteration steps by width of image // selection statement inside loop to execute if first pixel in a row. 1 mark: Selection statement checking pixel data at loop position. 1 mark: Test against one of the pixel values (0-4). 1 mark: Tests covering all pixel values. 1 mark: String has a correct character added to it for one of the pixel values. 1 mark: String has correct characters added to it for all of the pixel values. 1 mark: The hidden string is output. 1 mark: any other message Max 9 marks if any errors Python <pre>def decodeImage(width, image) : message = "" for position in range(0, len(image), width) : if image[position] == 0: message += "c" if image[position] == 1: message += "k" if image[position] == 2: message += "l" if image[position] == 3: message += "o" if image[position] == 4: message += " " print(f"The message is {message}")</pre> C# <pre>static void decodeImage(int width, int[] image) { string message = ""; for (int position = 0; position < image.Length; position += width) { switch (image[position]) { case 0:</pre>	10 AO3=10

		<pre> message += "c"; break; case 1: message += "k"; break; case 2: message += "l"; break; case 3: message += "o"; break; case 4: message += " "; break; } } Console.WriteLine(\$"The message is {message}"); } </pre> VB <pre> Sub decodeImage(width As Integer, image As Integer()) Dim message As String = "" For position As Integer = 0 To image.Length - 1 Step width Select case image(position) Case 0: message = message + "c" Case 1: message = message + "k" Case 2: message = message + "l" Case 3: message = message + "o" Case 4: message = message + " " End Select Next Console.WriteLine(\$"The message is {message}") End Sub </pre>	
--	--	--	--

Question	Part	Marking guidance	Total marks
09	2	1 mark: Syntactically valid call to <code>decodeImage</code> after <code>displayImage</code> is called. Python <pre> match userChoice: case "1": width, height, image = loadImage() print(f"The image has size {width} by {height} pixels") case "2": displayImage(width, height, image) decodeImage(width, image) case "3": rleImage(width, height, image) case "4": width, height, image = stretchImage(width, height, image) case "5": image = lightenImage(image) case "6": image = darkenImage(image) case "Q": message = random.randint(0,1) if message == 0: print("Thanks for using this software") else: print("Goodbye") case _: print() print("You did not enter a valid option") </pre> C# <pre> switch (userChoice) { case "1": (width, height, image) = loadImage(); break; case "2": displayImage(width, height, image); decodeImage(width, image); break; </pre>	1 AO3=1

		<pre> case "3": rleImage(width, height, image); break; case "4": (width, height, image) = stretchImage(width, height, image); break; case "5": lightenImage(image); break; case "6": darkenImage(image); break; case "Q": if (message.Next(0, 2) == 0) { Console.WriteLine("Thanks for using this software"); } else { Console.WriteLine("Goodbye"); } break; default: Console.WriteLine(); Console.WriteLine("You did not enter a valid option"); break; } </pre> VB <pre> Select Case userChoice Case "1" Dim values As (width As Integer, height As Integer, image as Integer()) = loadImage() width = values.width height = values.height image = values.image Console.WriteLine(\$"The image has size {width} by {height} pixels") Case "2" displayImage(width, height, image) decodeImage(width, image) Case "3" </pre>	
--	--	--	--

		<pre> rleImage(width, height, image) Case "4" Dim values As (width As Integer, height As Integer, image as Integer()) = stretchImage(width, height, image) width = values.width height = values.height image = values.image case "5" lightenImage(image) Case "6" darkenImage(image) Case "Q" If message.Next(0, 2) = 0 Console.WriteLine("Thanks for using this software") Else Console.WriteLine("Goodbye") End If Case Else Console.WriteLine() Console.WriteLine("You did not enter a valid option") End Select </pre>	
--	--	---	--

Question	Part	Marking guidance	Total marks
09	3	Screen capture(s) must match code from Q9.1 & Q9.2 1 mark: Shows the message <code>look cool</code> <pre> 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: 1 Please enter the image name: pattern The image has size 5 by 9 pixels 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: 2   The message is <code>look cool</code> 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image Q..Quit Please enter a choice: </pre>	1 AO3=1

Question	Part	Marking guidance	Total marks
10	1	1 mark: Subroutine with correct parameters. l. other parameters 1 mark: Call to <code>createImage</code> 1 mark: Call to <code>createImage</code> has correct width. 1 mark: Call to <code>createImage</code> has correct height (2 x original). 1 mark: Iteration used to copy original pixel data. 1 mark: Pixel data stored in new image. 1 mark: Iteration used to copy original pixel data (for rotated image). 1 mark: Reversed pixel data is added to new image. 1 mark: Subroutine returns newly created image and height. l. subroutine also returns width Max 8 marks if any errors Python <pre>def extendImage(width, height, image): newHeight = height * 2 newImage = createImage(width, newHeight) for position in range(0, len(image)): newImage[position] = image[position] counter = len(image) for position in range(len(image)-1, 1, -1): newImage[counter] = image[position] counter+=1 return newHeight, newImage</pre> C# <pre>static (int, int[]) extendImage(int width, int height, int[] image) { int newHeight = height * 2; int[] newImage = createImage(width, newHeight); for (int position = 0; position < image.Length; position++) { newImage[position] = image[position]; } int counter = image.Length; for (int position = image.Length - 1; position > -1; position--) { newImage[counter] = image[position]; counter += 1; } return (newHeight, newImage);</pre>	9 AO3=9

		<pre> } VB Function extendImage(width As Integer, height As Integer, image as Integer()) As (Integer, Integer()) Dim newHeight = height * 2 Dim newImage As Integer() = createImage(width, newHeight) For position As Integer = 0 To image.Length - 1 newImage(position) = image(position) Next Dim counter As Integer = image.Length For position As Integer = image.Length - 1 To 0 Step -1 newImage(counter) = image(position) counter = counter + 1 Next return (newHeight, newImage) End Function </pre>	
--	--	--	--

Question	Part	Marking guidance	Total marks
----------	------	------------------	-------------

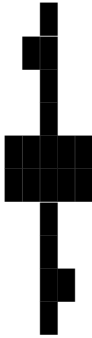
10	2	1 mark: Addition of option to indicate that 7 rotates image. Python <pre> def displayMenu(): print() print("1..Load image") print("2..Display image") print("3..Run length encode image") print("4..Stretch image") print("5..Lighten image") print("6..Darken image") print("7..Extend image") print("Q..Quit") print() </pre> C# <pre> static void displayMenu() { Console.WriteLine(); Console.WriteLine("1..Load image"); Console.WriteLine("2..Display image"); Console.WriteLine("3..Run length encode image"); </pre>	1 AO3=1
----	---	---	-------------------------------------

		<pre> Console.WriteLine("4..Stretch image"); Console.WriteLine("5..Lighten image"); Console.WriteLine("6..Darken image"); Console.WriteLine("7..Extend image"); Console.WriteLine("Q..Quit"); } </pre> VB <pre> Sub displayMenu() Console.WriteLine() Console.WriteLine("1..Load image") Console.WriteLine("2..Display image") Console.WriteLine("3..Run length encode image") Console.WriteLine("4..Stretch image") Console.WriteLine("5..Lighten image") Console.WriteLine("6..Darken image") Console.WriteLine("7..Extend image") Console.WriteLine("Q..Quit") End Sub </pre>	
--	--	--	--

Question	Part	Marking guidance	Total marks
10	3	1 mark: Selection structure detects input of 7 1 mark: Syntactically valid call to <code>extendImage</code> with return being stored. 1 mark: <code>height</code>, <code>image</code> variables are updated appropriately. 1. <code>width</code> also updated Python <pre> match userChoice: case "1": width, height, image = loadImage() print(f"The image has size {width} by {height} pixels") case "2": displayImage(width, height, image) decodeImage(width, image) case "3": rleImage(width, height, image) case "4": width, height, image = stretchImage(width, height, image) case "5": image = lightenImage(image) case "6": image = darkenImage(image) case "7": height, image = extendImage(width, height, image) case "Q": message = random.randint(0,1) if message == 0: print("Thanks for using this software") else: print("Goodbye") case _: print() print("You did not enter a valid option") </pre> C# <pre> switch (userChoice) { case "1": (width, height, image) = loadImage(); break; case "2": </pre>	3 AO3=3

		<pre> displayImage(width, height, image); decodeImage(width, image); break; case "3": rleImage(width, height, image); break; case "4": (width, height, image) = stretchImage(width, height, image); break; case "5": lightenImage(image); break; case "6": darkenImage(image); break; case "7": (height, image) = extendImage(width, height, image); break; case "Q": if (message.Next(0, 2) == 0) { Console.WriteLine("Thanks for using this software"); } else { Console.WriteLine("Goodbye"); } break; default: Console.WriteLine(); Console.WriteLine("You did not enter a valid option"); break; } </pre> VB <pre> Select Case userChoice Case "1" Dim values As (width As Integer, height As Integer, image as Integer()) = loadImage() width = values.width height = values.height image = values.image 'QN </pre>	
--	--	---	--

		<pre> Console.WriteLine(\$"The image has size {width} by {height} pixels") Case "2" displayImage(width, height, image) decodeImage(width, image) Case "3" rleImage(width, height, image) Case "4" Dim values As (width As Integer, height As Integer, image as Integer()) = stretchImage(width, height, image) width = values.width height = values.height image = values.image case "5" lightenImage(image) Case "6" darkenImage(image) Case "7" Dim values As (height As Integer, image As Integer()) = extendImage(width, height, image) height = values.height image = values.image Case "Q" If message.Next(0, 2) = 0 Console.WriteLine("Thanks for using this software") Else Console.WriteLine("Goodbye") End If Case Else Console.WriteLine() Console.WriteLine("You did not enter a valid option") End Select </pre>	
--	--	--	--

Question	Part	Marking guidance	Total marks
10	4	Screen capture(s) must match code from Q10.1, Q10.2 & Q10.3. 1 mark: Image is displayed with a rotated version of itself underneath. <pre> 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image 7..Extend image Q..Quit Please enter a choice: 1 Please enter the image name: one The image has size 5 by 5 pixels 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image 7..Extend image Q..Quit Please enter a choice: 7 1..Load image 2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image 7..Extend image Q..Quit Please enter a choice: 2  The message is cccc cccc 1..Load image </pre>	1 A03=1

		<pre>2..Display image 3..Run length encode image 4..Stretch image 5..Lighten image 6..Darken image 7..Extend image Q..Quit Please enter a choice:</pre>	
--	--	--	--