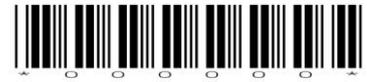
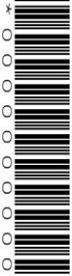


## ENTRY LEVEL CERTIFICATE IN COMPUTING

R353/XX: Programming



### INSTRUCTIONS TO TEACHERS

- Please refer to Section 4 of the specification for instructions on completing internal assessment tasks
- Each task can be contextualised appropriately to suit facilities available in your centre
- The marking criteria (Appendix A in specification) should be available whilst completing the tasks
- The total number of marks for this unit is **40**.

### INFORMATION FOR CANDIDATES

- This document consists of **7** pages. Any blank pages are indicated.



## Specimen Task A (text based languages)

A company produces games to run on digital television sets. You have been asked to write a quiz program for them. The quiz can be on a theme of your choice. Where you are asked to give evidence of your program you should print out or produce a screenshot of your program. You should check your program works at every stage but only need to give proof of testing in section (i).

Your program will:

- Start by asking for your name.
  - Welcome you to the quiz by name.
  - Ask five multiple choice questions.
  - For each question, if the player has the correct answer, add one to their score.
  - At the end of the quiz display the player's name and score .
  - Display Well Done 5 times if a player has a score of more than 3.
- (a) List what your quiz needs to be able to do. Plan the quiz program you are going to make. (You can use flowcharts, pseudo code or any other sensible method of showing how your program will work.)
- (b) Write a program to ask the player their name and then welcome them to the quiz. For example:

```
What is your name? Jerry  
Hello, Jerry welcome to the cheese quiz
```

Test your program works then give evidence of your code.

- (c) Add code to your program so asks a question and takes in an answer:

For example:

```
Question One: Which of the following is a goat cheese?  
A) Cheddar  
B) Caprino  
C) Stilton  
D) Brie  
Enter your answer B
```

Test your program works then give evidence of your code.

- (d) Improve your program so it adds one to the score if the player gets the answer right.

Test your program works then give evidence of your code.

- (e) Add four more questions to your quiz.

Test your program works then give evidence of your code.

- (f) When the quiz is over add code so it prints out the player's score.

```
Jerry has a score of: 5
```

Test your program works then give evidence of your code.

**(g)** If the player has a score of greater than three display: Well Done

Well Done

Test your program works then give evidence of your code.

**(h)** Change your program so it says “well done” 5 times if the player has a score of greater than 3.

Well Done

Well Done

Well Done

Well Done

Well Done

Test your program works then give evidence of your code.

**(i)** Test your program works. Make notes on how you tested it and what you found out.

**(j)** Write an evaluation of how well your program works. You should include:

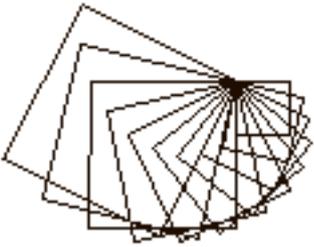
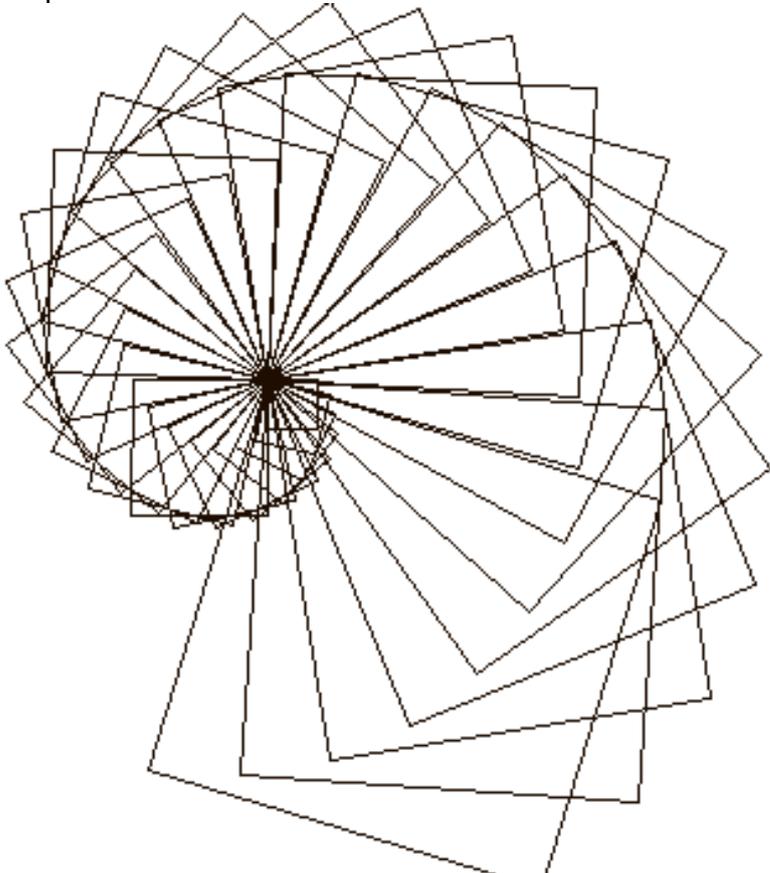
- Any problems you had making it and how you overcame them.
- Any bugs your program still has.
- Any future improvements that could be made to your program.

### Specimen Task - *Scratch*

A square of sides 10 can be drawn with a pen using the following instructions:

- Move forward 10
- Turn right 90 degrees
- Move forward 10
- Turn right 90 degrees
- Move forward 10
- Turn right 90 degrees
- Move forward 10
- Turn right 90 degrees

By repeating the process of drawing a square, turning slightly and drawing a bigger square it is possible to draw patterns.

<p>1 Square</p> 	<p>3 Squares</p> 
<p>10 Squares</p> 	<p>27 Squares</p> 

A printing company wants to draw these patterns so they can put them on T-Shirts. You are going to write a drawing program to create patterns for the company,

Where you are asked to give evidence of your program you should print out or produce a screenshot of your program. You should check your program works at every stage but only need to give proof of testing in section (i).

Your program will:

- Ask how many squares to draw.
- If the user chooses a number less than 15 picks a red pen otherwise black.
- Use this to answer to draw this many squares in the pattern previously described.

**(a)** List what your drawing program must be able to do. Make a plan for the drawing program. (You can use flowcharts, pseudo code or any other sensible method of showing how your program will work.)

**(b)** Write a program that selects a black coloured pen.

**(c)** Add to the program so that after selecting the pen colour it then draws a square with sides of length 20 and returns to its original position. (A reminder the steps for doing this are shown on the previous page.)

Test what you have made works then give evidence of your program blocks.

**(d)** After drawing the square make the sprite turn 13 degrees.

Test what you have made works then give evidence of your program blocks.

**(e)** Add to the program so it repeats the process of drawing a square and turning 20 times.

Test what you have made works then give evidence of your program blocks.

**(f)** Add to the program so it increases the size of the sides of the square by 5 each time it draws a new one.

Test what you have made works then give evidence of your program blocks.

**(g)** Add to the program so it asks the user how many squares the user wants and then draws the squares that number of times.

Test what you have made works then give evidence of your program blocks.

**(h)** Change your program so when it asks for the number of squares if the user picks a number of less than 15 a red pen is selected.

Test what you have made works then give evidence of your program blocks.

**(i)** Test your program works. Make notes on how you tested it and what you found out.

**(j)** Write an evaluation of how well your program works. You should include:

- Any problems you had making it and how you overcame them.
- Any bugs your program still has.
- Any future improvements that could be made to your program.

**BLANK PAGE**

**PLEASE DO NOT WRITE ON THIS PAGE**



**Copyright Information:**

Permission to reproduce items where third-party owned material protected by copyright is included has been sought and, where possible, every reasonable effort has been made by the publisher (OCR) to trace copyright holders, but if any items have unwittingly been included, the publisher will be pleased to make amends at the earliest possible opportunity.

OCR is part of the Cambridge Assessment Group. Cambridge Assessment is the brand name of University of Cambridge Local Examinations Syndicate (UCLES), which is itself a department of the University of Cambridge.