Instructions to Candidates

This paper will be marked out of 100 marks. It contributes 60% to the final mark for the unit FIT1040. The paper is divided into 4 parts labelled A, B, C and D.

Part A contains multiple-choice questions and is worth 20 marks. Each question in part A is worth 1 mark. There are a total of 20 questions. Marks are awarded for each correct answer. No mark is awarded (or deducted) for incorrect answers.

Part B contains 5 questions that will require short answers and is worth 25 marks. Most questions will require you to read and interpret Scribble code. Each question in this section is worth 5 marks.

Part C contains 3 questions and is worth 30 marks. Each question will require you write a short Scribble script. Each question in this section is worth 10 marks.

Part D contains a single question worth 25 marks. The question requires that you write a description of a complete Scribble application including all the scripts.

Three reference guides have been included within the paper to assist you in answering the questions in the paper. The first shows the flowchart symbols to be used in the paper. The second provides a list of the Scribble code blocks. The third provides a guide to handwriting Scribble code.

Answer all questions in the spaces provided within this examination paper. Spare space is provided at the end of the paper that can be used if the space required to answer a question exceeds the space available. Please use a pen (other than red) to write your answers.

Flow Chart Symbols

Please use these symbols when drawing flowcharts in this paper.
Scribble Code Block Reference Guide

**Motion Code Blocks**

- `move 10 steps`
- `turn ◄ 15 degrees`
- `turn ◄ 15 degrees`
- `point in direction 90 ◄`
- `point towards ◄`
- `go to x: -170 y: -83`
- `go to ◄`
- `go to random location`
- `glide 1 secs to x: -170 y: -83`
- `change x by 10`
- `set x to 0`
- `change y by 10`
- `set y to 0`
- `if on edge, bounce`

**Looks Code Blocks**

- `switch to costume costume 1`
- `next costume`
- `say Hello! for 2 secs`
- `say Hello!`
- `think Hmm... for 2 secs`
- `think Hmm...`
- `change color effect by 25`
- `set color effect to 0`
- `clear graphic effects`
- `change size by 10`
- `set size to 100 %`
- `size`
- `show`
- `hide`
- `go to front`
- `go back 1 layers`
Control Code Blocks

when clicked

when space key pressed

when Sprite1 clicked

wait 1 secs

forever

repeat 10

broadcast and wait

when I receive

forever if

if
Sensing Code Blocks

touching ?
touching color ?
color is touching ?

ask What's your name? and wait

answer

mouse x
mouse y
mouse down?

key space pressed?

distance to

reset timer

timer

x position of Sprite1

loudness

loud?

slider sensor value

sensor button pressed ?

object

attribute

Sound Code Blocks

play sound

play sound until done

stop all sounds

play drum 48 for 0.2 beats

rest for 0.2 beats

play note 60 for 0.5 beats

set instrument to 1

change volume by -10

set volume to 100 %

volume

change tempo by 20

set tempo to 60 bpm

tempo
Text Code Blocks

clear

write

set font to Aardvark

set font size to 12

change font color by 10

set font color to

change font shade by 10

set font shade to 50
Handwriting *Scribble* Code

In this paper, when hand writing *Scribble* code, simply write the words used to identify each individual code block on a separate line. Any sprites, costumes, lists or other variables used should be listed and described in the answer space provided. It is a good idea to show parameters passed to code blocks inside brackets (). The beginning and ending of control code blocks, such as a branch of an if-then code block or a looping code block should be visually shown in the manner used in the example below.

**Example Scribble Code**

![Scribble code example]

**Handwritten equivalent**

```
There are three variables used in this script. *steps* and *angle* are local to the sprite and contain numbers. *draw* is a global variable and contains a Boolean value.

when green flag clicked
pen down
set (steps) to (1)
set (angle) to (pick a random (10) to (100))
repeat (200)
if (draw) = <true>
  move (steps) *steps*
turn (angle) *degrees*
if on edge, bounce
change (steps) by (1)
change pen color by (pick random (1) to (10))
change pen shade by (pick random (1) to (10))
else
  say (I'm not ready)
```

pen up