



UNIVERSITY OF CAMBRIDGE INTERNATIONAL EXAMINATIONS
 General Certificate of Education
 Advanced Subsidiary Level and Advanced Level

CANDIDATE
 NAME

CENTRE
 NUMBER

--	--	--	--	--

CANDIDATE
 NUMBER

--	--	--	--

* 9 2 0 1 7 4 3 0 3 2 *

COMPUTING

9691/23

Paper 2

May/June 2011

2 hours

Candidates answer on the Question Paper.

No additional materials are required.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name on all the work you hand in.

Write in dark blue or black pen.

You may use a soft pencil for any diagrams, graphs or rough working.

Do not use staples, paper clips, highlighters, glue or correction fluid.

DO NOT WRITE IN ANY BARCODES.

Answer **all** questions.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [] at the end of each question or part question.


This document consists of **11** printed pages and **1** blank page.



- 1 Jodelle needs to write program code that will check the password to her personal computer. It checks each attempt to enter the password and closes the screen after three wrong attempts.
She wants the log-in screen to display:

- a request to enter the password
- space to enter the password
- how many attempts have been made
- a message if the log-in has been unsuccessful
- a means of returning to the previous screen

(a) Draw a suitable layout for the screen.



[5]

Jodelle first produces her solution using pseudocode. She wants the password to be 'poppy', the name of her cat.

```
Attempt ← 1
REPEAT
    INPUT Password
    Attempt ← Attempt +1
UNTIL Password = "poppy" OR Attempt = 3
IF Password = "poppy"
    THEN
        OUTPUT "password correct"
    ELSE
        OUTPUT "no valid password entered"
ENDIF
```

Jodelle needs to check whether this pseudocode works.

For
Examiner's
Use

(b) Complete the final row of the trace table for this pseudocode using poppy as input.

Attempt	Password	Password = "poppy"	Attempt = 3	Password = "poppy" OR Attempt = 3	Output
1					
	poppy				
2					
		True			
			False		
				True	

[1]

(c) Complete the trace table for this pseudocode code using cat as first input, followed by poppy as second input.

Attempt	Password	Password = "poppy"	Attempt = 3	Password = "poppy" OR Attempt = 3	Output
1					

[6]

(d) There is an error in this pseudocode.

(i) One way to correct this is to change the first line. Rewrite this line to make the pseudocode correct.

..... [1]

(ii) State the type of error that Jodelle made.

..... [1]

(e) Using the name of Jodelle's cat was not a good choice of password.

(i) Give **two** features to consider when choosing a password which gives greater security.

1

.....

2

..... [2]

(ii) Suggest a better password that Jodelle could use.

..... [1]

(f) Jodelle wants to output a comment after each failed password entry.
If Attempt = 1 she wants to output "First try is wrong. Please try again"
On the second try she wants to output "Password still wrong. One more chance"
On the third try she wants to output "No valid password entered"

In a high-level language write a program for this idea using a CASE or SELECT statement.

Language

Code

.....

.....

.....

.....

..... [4]

2 Amber stores the names of her favourite song tracks in an array. She has 56 track names at the moment and expects to reach 150. She defined the array as one-dimensional, size 150 and will store strings. She then initialised each element of the array.

(a) (i) What value should Amber give each element?

..... [1]

(ii) Define the array and initialise each element of the array using a high-level programming language of your choice.

Language

Code

.....
.....
.....
.....
.....
.....
.....
.....
..... [4]

(b) Write additional code which will search the contents of the array for a particular track name. Output the position of this track name in the array. (You may assume the track name exists in the array).

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
..... [5]

- (c) As her collection grows, Amber decides to hold more information about each track. The data for each track is structured as a record.

*For
Examiner's
Use*

Each record contains the following data:

- unique track ID (a whole number between 1 and 150)
- track name
- date bought
- cost
- solo artist (yes/no)

Complete the following table.

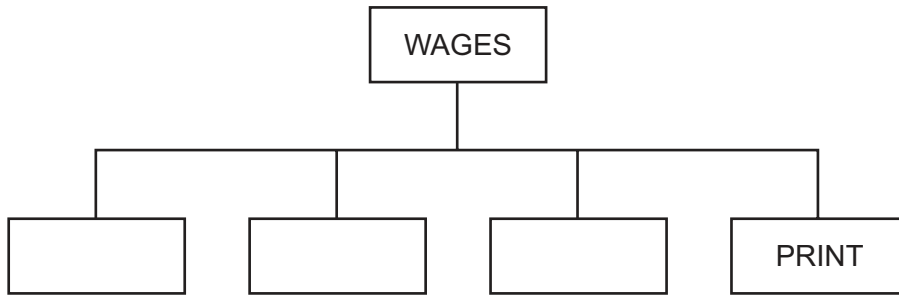
Field Name	Data Type	Size of Field (bytes)
TrackID		
TrackName		
DateBought		
Cost		
SoloArtist		

[10]

3 Kris has written a program that will work out the wages for her staff. The main steps for each employee are: to work out the hours worked, work out the total earnings, work out tax and finally print out how much will be taken home.

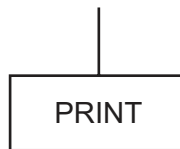
For
Examiner's
Use

(a) Complete the structure diagram to show the modules that will be needed.



[3]

(b) The printout will be different for those staff who receive cash and those who have their earnings paid directly to a bank account. Add the next level to the print module.



[2]

(c) Describe **two** programming techniques that should be used to help other programmers maintain the programming code.

Technique 1

.....

.....

.....

Technique 2

.....

.....

.....

[4]

- (d) For each employee the hours worked module collects data for five days. If they do not work on a particular day a zero is entered. Each person can work up to 9 hours a day for up to 5 days a week. The hours are added up; no-one may work more than 40 hours.

Write **five** sets of test data which test the module for different inputs/outcomes. For each set of test data, give a reason for your choice.

	Day 1	Day 2	Day 3	Day 4	Day 5
Test 1					
Reason					
Test 2					
Reason					
Test 3					
Reason					
Test 4					
Reason					
Test 5					
Reason					

[10]

For
Examiner's
Use

(e) (i) Explain how it is possible to use the same variable name for different variables in different modules.

.....
.....
.....
..... [2]

(ii) Explain how this can help when modules are programmed separately.

.....
.....
.....
.....
.....
..... [3]

*For
Examiner's
Use*

4 The following pseudocode is a recursive function where n is an integer.

```
FUNCTION calc(n)
  IF n = 1
    THEN
      calc ← 1
    ELSE
      calc ← n * calc(n-1)
  ENDIF
  RETURN
```

(a) (i) What value is returned by calc(1)?

..... [1]

(ii) What value is returned by calc(3)?

..... [1]

(b) (i) What happens if the parameter passed is -1?

.....
.....
.....
..... [2]

(ii) What changes will need to be made to the pseudocode to address the problem in (b)(i)?

.....
.....
.....
..... [2]

(c) Rewrite this function in pseudocode as an iterative function.

.....
.....
.....
.....
.....
.....
.....
.....
..... [4]

*For
Examiner's
Use*

BLANK PAGE

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

University of Cambridge International Examinations 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.