CAMBRIDGE INTERNATIONAL EXAMINATIONS

Cambridge Ordinary Level

MARK SCHEME for the October/November 2014 series

7010 COMPUTER STUDIES

7010/12 Paper 1, maximum raw mark 100

This mark scheme is published as an aid to teachers and candidates, to indicate the requirements of the examination. It shows the basis on which Examiners were instructed to award marks. It does not indicate the details of the discussions that took place at an Examiners' meeting before marking began, which would have considered the acceptability of alternative answers.

Mark schemes should be read in conjunction with the question paper and the Principal Examiner Report for Teachers.

Cambridge will not enter into discussions about these mark schemes.

Cambridge is publishing the mark schemes for the October/November 2014 series for most Cambridge IGCSE®, Cambridge International A and AS Level components and some Cambridge O Level components.

CAMBRIDGE
International Examinations

1 Any **three** from:

- (provides) user interface
- input/output control
- security
- handling interrupts
- spooling
- memory management
- processor management
- utilities (e.g. copy, save, delete, re-name, etc.)
- maintain user accounts
- load/run software
- error reporting/handling
- multiprogramming
- batch processing (JCL)/real time processing
- multitasking/multiuser/multi-access
- file management [3]

2 **(a)** 1 mark for way + 1 mark for reason

way	impact
– deskilling	– software has removed the need for some of the more traditional skills e.g. using CAD
– (re-)training	– work practices have changed; need to learn how to use the new software/computer
– redundancy	– new technology allows work to be completed by fewer staff/out-sourced to "cheaper" work forces in foreign countries
– work from home (etc.)	– use of emails, VoIP, video conferencing, instant messaging, etc. allows working away from the office
– nicer/safer work place	– quieter (no noisy typewriters) and safer (no heavy filing cabinets)
– health (& safety)	– RSI, headaches, backaches

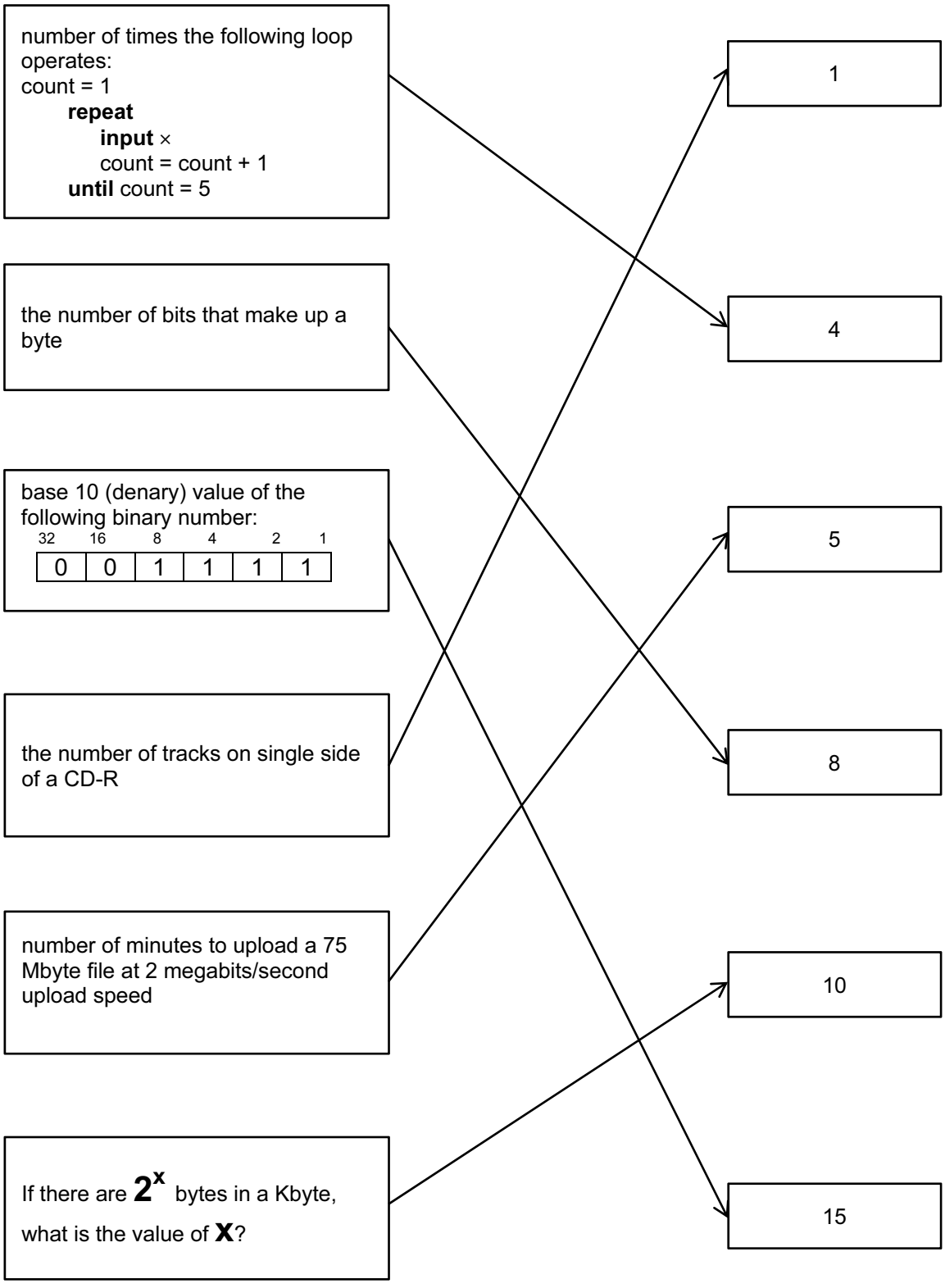
[6]

(b) Any **two** health risks from:

- RSI in wrists or fingers from prolonged typing or repeated clicking of mouse button
- headaches/eyestrain/dry eye from staring at a monitor for long periods of time/glare from monitor
- back ache/strain from sitting in same position for a long time/using chair with no adjustment [2]

3 **(a)** hacking [1]
(b) viruses [1]
(c) phishing [1]
(d) spyware [1]
(e) pharming [1]

4 1 mark for each correct connection up to the maximum of 5



[5]

5 (a)

Sat Nav devices send signals to the global positioning satellites	***TRUE***	***FALSE*** (oval)
Sat Nav accurately measures vehicle speed using satellite position and accurate timing	***TRUE*** (oval)	***FALSE***
Satellites tell the Sat Nav which direction the vehicle should take	***TRUE***	***FALSE*** (oval)

[3]

(b) Any **two** from:

– software/maps not up to date/new road
– loss of satellite signals
– wrong data input by user (e.g. start point and end point) [2]

6 (a) 1 mark for each error and suggested correction (accept description or example of corrected pseudocode).

error: line 10: total = 1
correction: totals should be set to zero; total = 0

error: line 30: … number < 10 …
correction: check should be made if number > 10; … number > 10 …

error: no input inside loop
correction: **input** number

error: line 50: x = x + 1
correction: **for … to** loops don't need a counter; remove line 50 altogether

error: line 80: **output** x
correction: output should be total value; **output** total [5]

(b) division by zero error (or similar description of error produced when dividing by 0)

add an error trap after input of number
e.g. 40 **if** number = 0 **then** k = 0 **else** k = x/number [2]

7 (a) (i) – higher quality photos
– when "blown up" less likely for photo to "pixelate" [1]

(ii) – uses up more memory (on card)
– takes longer to upload/download a photo
– file size will be greater
[1]

(b) (i) – solid state memory
– flash drive
– non-volatile [1]

(ii) – no moving parts (so more robust)
– can be removed from camera and retain its contents
– can erase contents and reuse memory card [1]

(c) (i) – **(pic)**ture **(el)**ement [1]

(ii) – **819 or 1638** [1]

(d) Any **one** point from: e.g.

– auto flash
– anti (hand) shake facility
– easy deletion of unwanted photos
– ability to "manipulate" images after they have been taken/special effects
– "smart" operation e.g. automatically pick out objects, faces, etc.
– auto capture [1]

8 (a) Any **one** from:

– infra red (sensor)
– pressure (sensor)
– proximity (sensor) [1]

(b) Any **one** from:

– additional sensors used
– door defaults to open position
– sounds an alarm if a sensor fails [1]

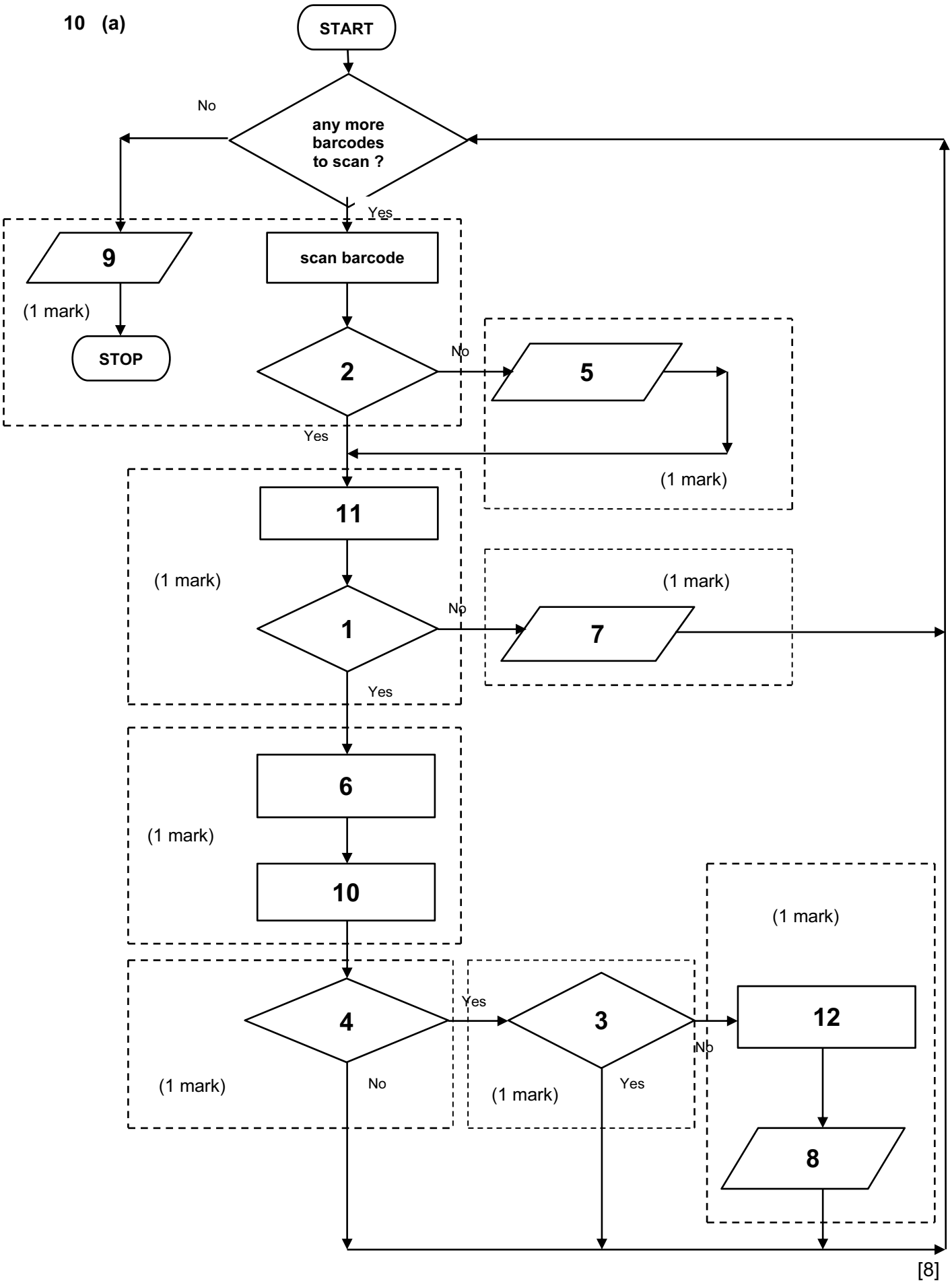
(c) Any **four** points from:
- sensors continuously send signals/data
- sensor sends signals/data sent to the microprocessor
- signal converted to digital if necessary (using ADC)
- microprocessor checks which door(s) is (are) affected
- microprocessor compares sensor reading with stored values
- if reading indicates passenger detected…
- …microprocessor sends signal/data to actuators/motor…
- (converted to analogue using DAC)
- …to operate motors to open doors
- microprocessor also send signal to driver's cab (automatically) to sound an alarm
- monitoring continues until system switched off [4]

9

```
1  PENDOWN
2  LEFT 90
3  REPEAT 2
4  FORWARD 20                    1 mark
5  RIGHT 90
6  ENDREPEAT
   --------------------
7  FORWARD 20
8  LEFT 90
9  FORWARD 20                    1 mark
10 LEFT 90
   --------------------
11 FORWARD 20
12 RIGHT 90                      1 mark
13 FORWARD 20
   --------------------
14 RIGHT 90
15 FORWARD 20                    1 mark
16 PENUP
   --------------------
17 FORWARD 20
18 PENDOWN
19 FORWARD 20                    1 mark
20 RIGHT 90
   --------------------
21 FORWARD 60
22 RIGHT 90                      1 mark
23 FORWARD 20
```

[NOTE: – award 1 mark for each correct block (shown separated by dotted lines)
– look out for alternative solutions using REPEAT/ENDREPEAT which may be correct
– if a mistake in one of the blocks, start marking from the end awarding marks for correct blocks up to the error] [6]

10 (a)



[8]

(b) 1 mark for each device + 1 mark for correct matching use

device: beeper/loud speaker
use: to indicate barcode correctly read/error in reading barcode

device: (LCD) screen/monitor
use: to show prices and other information about goods

device: touch screen
use: to show prices and other information about goods/to select items that need to be weighed/identified

device: weighing machine
use: to find weight of loose items (e.g. fruit) to enable pricing

device: (magnetic) card reader/CHIP and PIN reader
use: to read customer's debit/credit card/enable customer to pay for goods using a credit or debit card

device: printer
use: to print receipts

(NOT keypad) [4]

11 (a) = (B2/24) * B3 [1]

(b) = C4 * B5 [1]

(c) = IF (D1 < C6, "profit", "no profit") (NOTE: accept C6 > D1 in formula)
1 mark 1 mark [2]

(d)

	A	B	C	D
1				1500
2		18		
3		60		
4			45	
5		40		
6			1800	profit
		1 mark	1 mark	1 mark

[3]

12 (a) 1 mark for each of four rows shown in bold below; there are two possible ways of doing this – one set of answers is shown on the left and the alternative is shown on the right. Don't allow mix and match; answers must either be as shown on the left OR as shown on the right

0 0 0 0 0 0

0 0 1 0 0 1 **(OR 0 0 1 1 0 1)** – 1 mark

0 0 0 0 0 1 **(OR 0 0 0 1 0 1)** – 1 mark

0 0 0 0 0 0

0 0 0 0 0 0

0 0 1 1 0 1 **(OR 0 0 1 1 0 0)** – 1 mark

0 0 0 1 0 1 **(OR 0 0 0 1 0 0)** – 1 mark

0 0 0 0 0 0 [4]

(b) 2 marks for identifying the letter

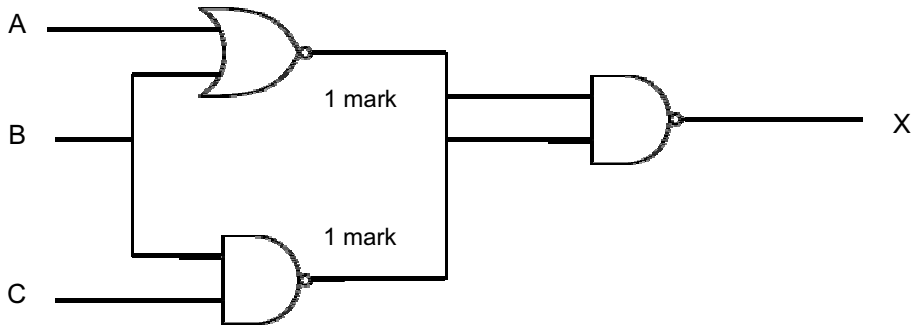
letter: H [2]

13 (a)

A	B	C	X	
0	0	0	**0**	1 mark
0	0	1	**0**	
0	1	0	**1**	1 mark
0	1	1	**1**	
1	0	0	**1**	1 mark
1	0	1	**1**	
1	1	0	**1**	1 mark
1	1	1	**1**	

[4]

(b) 1 mark per correct NOR gate and NAND gate on the left **(ONLY accept two-input gates)**



[2]

(c) 1 mark per logic statement as shown below:

(A = 1 **AND** B = 1) **OR** (B = **NOT** 1 **AND** C = 1)
(1 mark) (1 mark) (1 mark)

The above can be written as:
(A **AND** B) **OR** (**NOT** B **AND** C)
(1 mark) (1 mark) (1 mark)

Note: allow 1st part of formula and 2nd part of formula to be reversed:
(e.g. (NOT B AND C) OR (A AND B))

Also accept Boolean algebra:
a.b + $\bar{b}$.c (can be written as: A.B + B.C)
(1 mark) (1 mark) (1 mark)

[3]

14 **NOTE:** sum1, sum2 and total MUST be initialised for all three inputs to get the mark; allow repetition in any of the columns EXCEPT the OUTPUT column (e.g. sum1 can be 0, 47, 47, 47, 47, 47, 47);

sum1	sum2	total	a	b	c	d	e	f	OUTPUT
0	0	0	4	3	2	0	0	8	
47	8	55							
		44							
		33							
		22							
		11							
		0							data are OK
0	0	0	5	0	1	2	3	4	
34	16	50							
		39							
		28							
		17							
		6							
		-5							error
0	0	0	0	0	0	0	0	0	

1 mark 1 mark 1 mark < - - - - - - - - - - - - - - - - - 1 mark - - - - - - - - - - - - - - - - > 1 mark

[5]

15 1 mark per feature applied to text in question:

taxi ⟶ cabs – search and replace feature/retype/thesaurus

yelow ⟶ yellow – spell checker/retype/auto-correct

changed ⟶ replaced – search and replace feature/retype/thesaurus

translate last sentence/line – (auto-)translator English to Spanish/other language

[4]

16 marking points:

- initialise highest value (zero or less)
- loop control for all 3000 students
- set total = 0 (to find the average) before second loop
- loop control for all 8 exams
- check if input mark higher than stored highest mark
- if input mark higher, then set highest to this new value
- find the average mark for each student (includes correct total addition)
- both outputs in the correct place
  (average after inside loop, highest outside outer loop)
  (must be an attempt to find both average and highest to earn this mark)

sample program:

```
highest = -1                                    1 mark
for student = 1 to 3000                         1 mark
    total = 0                                   1 mark
    for exam = 1 to 8                           1 mark
        input mark
        total = total + mark
        if mark > highest then highest = mark   2 marks
    next
    average = total/8                           1 mark
    output average
next
output highest                                  1 mark
```

[5]