AQA

GCSE

Computer Science

Computing Fundamentals – 4512/2

Mark scheme

4512
June 2015

Version 1: Final Mark Scheme

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 Assessment Writer.

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 aqa.org.uk

<table>
<tr><th>Qu</th><th>Part</th><th>Sub-part</th><th>Marking Guidance</th><th>Marks</th></tr>
<tr><td>1</td><td>a</td><td></td><td>186;</td><td>1</td></tr>
<tr><td>1</td><td>b</td><td></td><td>E;</td><td>1</td></tr>
<tr><td>1</td><td>c</td><td></td><td>76;

If the answer given is 76 then reward any attempt at working;
If the answer given is not 76 then a maximum of 1 mark can be awarded for any of the following working out stages:

• Show multiplication of 4 by 16 and another number between 0 and 16 by 1 (i.e. allow C to be incorrectly converted to decimal).
• Convert to binary 1001100 but then incorrectly converted to denary // convert to binary 01001100 but then incorrectly converted to denary.
• Convert to a binary number other than 1001100, which must consist of more than 4 bits, but then convert this binary number to its correct decimal representation.</td><td>2</td></tr>
<tr><td>1</td><td>d</td><td></td><td>1 mark if 1 number correct;
2 marks if 2 numbers correct;
3 marks if all 4 numbers correct;
The correct order is: 2, 1, 3, 4

<table><tr><th>Quantity</th><th>Order (1-4)</th></tr><tr><td>15 bits</td><td>2</td></tr><tr><td>3 nibbles</td><td>1</td></tr><tr><td>2 bytes</td><td>3</td></tr><tr><td>1 kilobyte</td><td>4</td></tr></table></td><td>3</td></tr>
<tr><td>1</td><td>e</td><td></td><td>110 0111;

R. if more than 7 bits used (eg 0110 0111)</td><td>1</td></tr>
<tr><td>1</td><td>f</td><td>i</td><td>C;

(correct answer only, do not award if more than one box is ticked)</td><td>1</td></tr>
<tr><td>1</td><td>f</td><td>ii</td><td>$4//2^2$;</td><td>1</td></tr>
<tr><td>1</td><td>f</td><td>iii</td><td>5;</td><td>1</td></tr>
<tr><td>1</td><td>f</td><td>iv</td><td>the resolution // number of pixels used // size of the grid // ppi (or equivalent) // compression;</td><td>1</td></tr>
</table>

			A. Any other sensible answer should be credited.	
2	**a**		No marks for the programming language alone. Any correct combination of data structure and language. Examples include: • Python and list • Python and dictionary • Java and array • C and struct **A.** Any of array//list//record//struct//class data structures if given without naming a programming language.	**1**
2	**b**		Any creditworthy point to a maximum of three. Examples include: The number of students may be unknown; A data structure will be easier to iterate over/traverse; A data structure could hold both the names and ages together; A data structure would make the program code easier to update/modify; Could use pre-written routines with a standard data structure; Could make it easier to reuse the code; **A.** examples such as 'easier to sort the data'.	**3**
3	**a**		A series of instructions//sequence of steps; (Designed to) perform a particular task//solve a problem; **A.** Other wording	**2**
3	**b**	**i**	It is an index//counter/stepper (for the array); **A.** Answer that refers to its role in array indexing such as "Used to show which item in the array is the current one."	**1**
3	**b**	**ii**	Boolean **A.** Bool (or similar abbreviation) **R.** True/False or Yes/No	**1**
3	**b**	**iii**	(The purpose of the algorithms is to) check if an array contains a specific value/the value "`diffie`"/the value of `a`;	**1**
3	**b**	**iv**	(Algorithm 2 is a better algorithm because) as soon as a match is made it stops (the while loop)//less matches need to be made//it is more efficient//it stops at the correct index//the value of `i` will be set to the index of the value `a` (`diffie`)	**1**

<table>
<tr><td>4</td><td>a</td><td></td><td>1 mark if 1 stage correct (and the candidate has written this stage only once);
2 marks if 2 stages correct (and the candidate has written both of these stages only once);
3 marks if all 4 stages correct;
The correct stages are:
• Stage 1 is Analysis
• Stage 2 is Design
• Stage 3 is Implementation
• Stage 4 is Testing</td><td>3</td></tr>
<tr><td>4</td><td>b</td><td>i</td><td>Spiral/iterative/agile/SCRUM/incremental/cyclic;

A. Any other correct model</td><td>1</td></tr>
<tr><td>4</td><td>b</td><td>ii</td><td>
<table>
<tr><td>No creditworthy material</td><td>0</td></tr>
<tr><td>Lower mark range
A correct description of prototyping is given.
//
A partial description of prototyping is given and one appropriate (dis)advantage is stated.
//
No creditworthy description of prototyping is given but one or two appropriate (dis)advantages are stated.

Quality of written communication: The candidate has used a form and style of writing which has many deficiencies. Ideas are not often clearly expressed. Sentences and paragraphs are often not well-connected or at times bullet points may have been used. Specialist vocabulary has been used inappropriately or not at all. Much of the text is legible and some of the meaning is clear. There are many errors of spelling, punctuation and grammar but it should still be possible to understand much of the response.</td><td>1-2 marks</td></tr>
</table>
</td><td>6</td></tr>
</table>

				Mid mark range A correct description of prototyping is given **and** one or two appropriate (dis)advantages are stated. // A partial description of prototyping is given **and** one or more (dis)advantages are described. Quality of written communication: The candidate has mostly used a form and style of writing appropriate to purpose and has expressed some complex ideas reasonably clearly and fluently. The candidate has usually used well linked sentences and paragraphs. Specialist vocabulary has been used on a number of occasions but not always appropriately. Text is legible and most of the meaning is clear. There are occasional errors of spelling, punctuation and grammar.	3-4 marks		
				High mark range A correct description of prototyping is given **and** two or more correct (dis)advantages are described. Quality of written communication: The candidate has selected and used a form and style of writing appropriate to purpose and has expressed complex ideas clearly and fluently. Sentences and paragraphs follow on from one another clearly and coherently. Specialist vocabulary has been used appropriately throughout. Text is legible and the meaning is clear. There are few if any errors of spelling, punctuation and grammar.	5-6 marks		
				Quality of written communication skills The candidate's quality of written communication skills will be one of the factors influencing the actual mark an examiner will give within a level of response. The quality of written communication skills associated with each level is indicated above.			

			Definition of prototyping (1 point is a 'partial' definition, whereas 2 points is a 'correct' definition): The construction of a simple/incomplete version of the complete program; that demonstrates how the program will work. **Examples of advantages of prototyping (any other correct advantage should be credited):** • Enable the system to be reviewed by the client/user during development. • May provide a final system that is better suited to the client/user's needs. • Will detect incorrect features earlier than other models. • Enable the developers to gain an early insight into how the system could be developed. **Examples of disadvantages of prototyping (any other correct disadvantage should be credited):** • Prototypes can be converted to final code that is hard to maintain. • Prototypes can be converted to final code of lower quality than a properly analysed solution.	
5	**a**	**i**	CountryName; **I** case, spaces and minor misspellings	**1**
5	**a**	**ii**	6;	**1**
5	**a**	**iii**	The primary key of the Country table; is also in/is the foreign key in the Airport table; // There is a field that appears in both tables; called CountryName;	**2**
5	**b**		third row only; **R**. if more than one box ticked	**1**

<table>
<tr><td>5</td><td>c</td><td></td><td>FCO, 4
FRA, 2
HAM, 2

1 mark for only displaying <code>Airport.Code</code> followed by <code>Airport.Terminals</code> (do not award if less or more fields are given);
1 mark for identifying the correct 3 records, even if the wrong fields are shown (i.e. the correct <code>Airport.Name</code> and/or <code>Airport.Code</code> must be present) (do not award if less or more records are given);
1 mark for results displaying in ascending alphabetic order (permit if other records incorrect as long as there are at least two records shown);</td><td>3</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>6</td><td>a</td><td></td><td>(A combination of) hardware and software;</td><td>1</td></tr>
<tr><td>6</td><td>b</td><td></td><td>One mark for each valid point below (maximum 4). If only one of memory or processor is referenced then maximum 3 marks.
The instructions are held in memory;
Loads instructions from secondary storage to memory;
Instructions are stored in a contiguous format;
The processor fetches an instruction from memory;
The processor decodes the instruction;
The processor executes the instruction;
The result may be stored back into memory;
The process is repeated continuously//cycles;

A. Any other correct answer</td><td>4</td></tr>
<tr><td>6</td><td>c</td><td></td><td>(Because the processor with two cores may be able to process) two instructions in parallel/at the same time/simultaneously;

A. Processing is shared.</td><td>1</td></tr>
<tr><td>6</td><td>d</td><td></td><td>The completed table is:
<table>
<tr><th>Description</th><th>Term</th></tr>
<tr><td>Uses a laser to read the data.</td><td>D (Optical media)</td></tr>
<tr><td>Contents cannot be edited.</td><td>E (ROM)</td></tr>
<tr><td>Small and very fast storage found close to the processor</td><td>A (Cache memory)</td></tr>
</table>
1 mark for each correct label.

A. The terms written out in full instead of the labels (do not penalise spelling errors)</td><td>3</td></tr>
</table>

<table>
<tr><td>6</td><td>e</td><td></td><td>(When a computer system has) insufficient memory;
When running a program / combination of programs too large to fit into the memory;
When the main memory is small / limited;
When the main memory is all in use;</td><td>1</td></tr>
<tr><td>7</td><td>a</td><td></td><td>RETURN;

Do not penalise spelling mistakes as long as the word is clear.</td><td>1</td></tr>
<tr><td>7</td><td>b</td><td></td><td>Real;

R. If more than one box ticked</td><td>1</td></tr>
<tr><td>7</td><td>c</td><td></td><td>1 mark for every correct row that appears in the correct sequence:

<table><tr><th>cost</th></tr><tr><td>0</td></tr><tr><td>2000</td></tr><tr><td>1000</td></tr><tr><td>500</td></tr></table>
Example answers and their associated marks are:

<table><tr><th>cost</th></tr><tr><td>500</td></tr><tr><td></td></tr><tr><td></td></tr><tr><td></td></tr></table>1 mark
<table><tr><th>cost</th></tr><tr><td>0</td></tr><tr><td>500</td></tr><tr><td></td></tr><tr><td></td></tr></table>2 marks</td><td>4</td></tr>
</table>

			cost 0 2000 1500 500 3 marks **cost** 2000 1000 500 3 marks	
7	**d**		1 mark for any correct answer to a maximum of 3. Answers include: It reduces repetition of code; It is easier to test; It is easier to maintain/update the program; It makes code more reusable; It makes code more elegant/understandable; It makes it easier for code to be developed in teams; It allows use of pre-written routines; It can speed up development time; **A.** Any other sensible answers.	**3**
8			**No creditworthy material** 0 **Lower mark range** 1-2 marks One or two (dis) advantages are stated. // One (dis)advantage is explained. Quality of written communication: The candidate has used a form and style of writing which has many deficiencies. Ideas are not often clearly expressed. Sentences and paragraphs are often not well-connected or at times bullet points may have been used. Specialist vocabulary has been used inappropriately or not at all. Much of the text is legible and some of the meaning is clear. There are many errors of spelling, punctuation and	**6**

				grammar but it should still be possible to understand much of the response.			
				Mid mark range Two or more (dis) advantages are explained Quality of written communication: The candidate has mostly used a form and style of writing appropriate to purpose and has expressed some complex ideas reasonably clearly and fluently. The candidate has usually used well linked sentences and paragraphs. Specialist vocabulary has been used on a number of occasions but not always appropriately. Text is legible and most of the meaning is clear. There are occasional errors of spelling, punctuation and grammar.	3-4 marks		
				High mark range Two or more (dis) advantages are discussed (including at least one advantage **and** one disadvantage). Quality of written communication: The candidate has selected and used a form and style of writing appropriate to purpose and has expressed complex ideas clearly and fluently. Sentences and paragraphs follow on from one another clearly and coherently. Specialist vocabulary has been used appropriately throughout. Text is legible and the meaning is clear. There are few if any errors of spelling, punctuation and grammar.	5-6 marks		
				Quality of written communication skills The candidate's quality of written communication skills will be one of the factors influencing the actual mark an examiner will give within a level of response. The quality of written communication skills associated with each level is indicated above.			

			Examples of advantages of connecting to a network (give credit to any other correct example): • Enables users to work from multiple physical locations. • Enables hardware resources to be shared between computers. • Enables computers to communicate with one another. • Creates more resilient systems (than when you are reliant on just one computer). • Enables processing to be distributed. • May enable access to web services. • Easier monitoring of all users. • Centralised back-up is possible. • Easier to maintain multiple devices. **Examples of disadvantages of connecting to a network (give credit to any other correct example):** • Additional hardware is required. • Introduces potential security risks. [allow a maximum of two points for viruses, hacking and so on] • Additional support costs • Certain hardware failures (e.g. main server or switch/router) could impact other devices • Performance potentially limited by network traffic.			

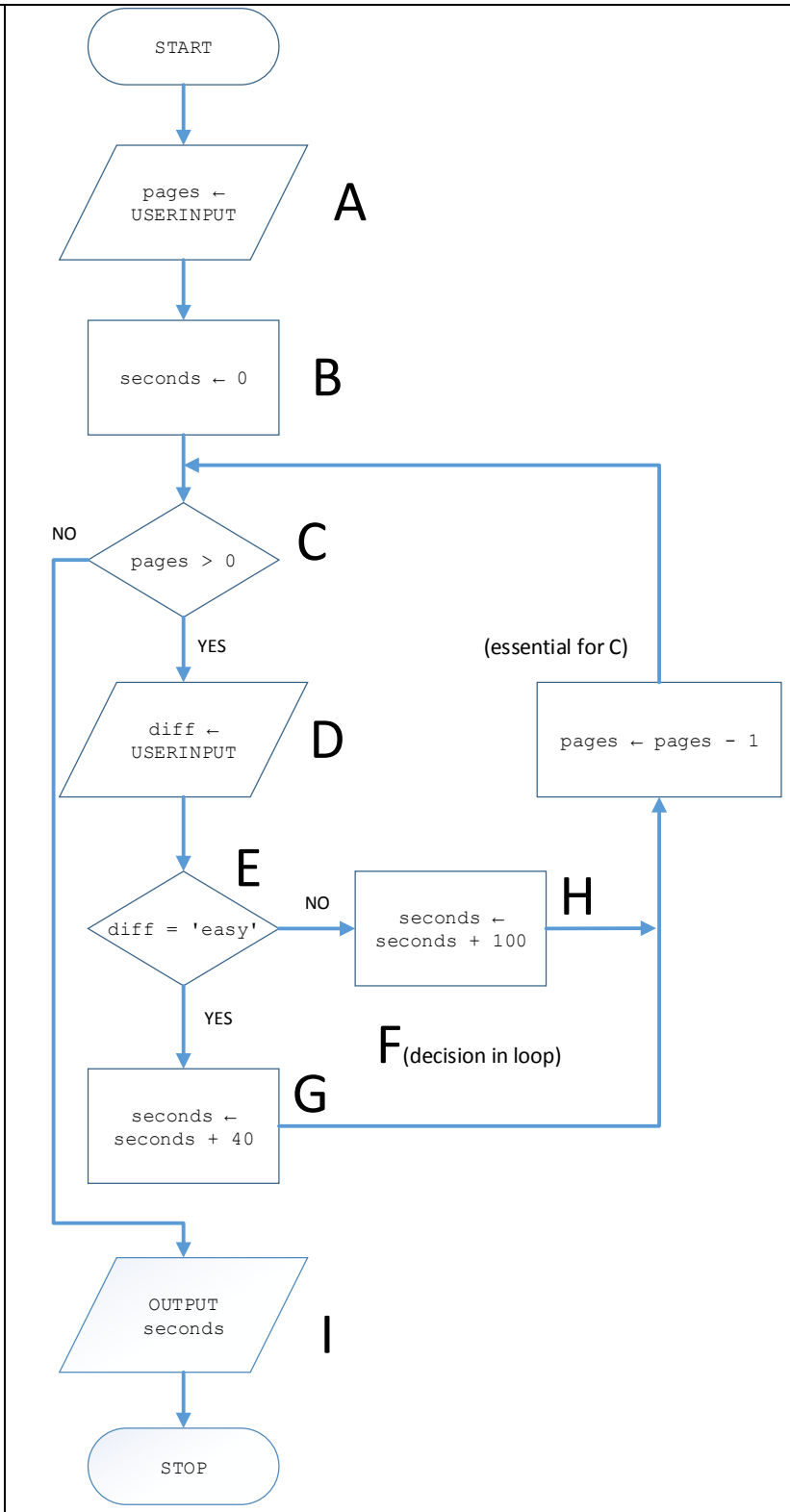
9			Marks awarded as follows (allow any logically equivalent and correct answer). The marks are labelled A – I and shown in the examples where they are awarded:
			A. **1 mark** for assigning user input to a variable (permit any variable name, `pages` has been used in the examples);
			B. **1 mark** for creating a variable that stores the total number of seconds (permit any variable name, `seconds` has been used in the examples) and instantiating this to zero (mark can only be awarded if this is declared **outside** of the loop);
			C. **1 mark** for using a loop to iterate over every page (two logically equivalent examples are given below although there are many logically equivalent ways to accomplish this);
			D. **1 mark** for asking for the user input for the page difficulty; ***(Note that no marks are awarded for validating the user input)***
			E. **1 mark** for using selection to decide if user input is 'easy' (this does not need to be explicit and could possibly the ELSE clause where the IF is asking if it is 'difficult');
			F. **1 mark** for using selection within the loop;
			G. **1 mark** for increasing the number of seconds by 40 within the correct selection block;
			H. **1 mark** for increasing the number of seconds by 100 within the correct selection block;
			I. **1 mark** for outputting the total number of seconds taken **outside** of the loop;
			
			Example 1 (every italicised square bracket indicates where that mark is awarded):
			`pages ← USERINPUT` *[A]*
			`seconds ← 0` *[B]*
			`REPEAT pages` *[C]*
			`  diff ← USERINPUT` *[D]*
			`  IF diff = 'easy' THEN` *[E][F as used within the loop]*
			`    seconds ← seconds + 40` *[G]*
			`  ELSE`
			`    seconds ← seconds + 100` *[H]*
			`  ENDIF`
			`ENDREPEAT`
			`OUTPUT seconds` *[I]*
			Example 2 (every italicised square bracket indicates where that mark is awarded):
			`pages ← USERINPUT` *[A]*
			`seconds ← 0` *[B]*
			`WHILE pages > 0` *[C]*
			`  diff ← USERINPUT` *[D]*
			`  IF diff = 'easy' THEN` *[E][F as used within the loop]*
			`    seconds ← seconds + 40` *[G]*
			`  ENDIF`
				9

```
  IF diff = 'difficult' THEN
    seconds ← seconds + 100 [H]
  ENDIF
  pages ← pages – 1 [essential for mark C]
ENDWHILE
OUTPUT seconds [I]
```

Example 3 (every italicised square bracket indicates where that mark is awarded):

```
pages ← USERINPUT [A]
seconds ← 0 [B]
FOR i ← 1 TO pages [C]
  IF USERINPUT = 'easy' THEN [D][E][F as used within the loop]
    seconds ← seconds + 40 [G]
  ELSE
    seconds ← seconds + 100 [H]
  ENDIF
ENDFOR
OUTPUT seconds [I]
```

Example 4 using a flowchart (large annotated letters indicate where that mark is awarded):


START
pages ← USERINPUT
A
seconds ← 0
B
NO
pages > 0
C
YES
(essential for C)
diff ← USERINPUT
D
pages ← pages - 1
E
diff = 'easy'
NO
seconds ← seconds + 100
H
YES
F (decision in loop)
seconds ← seconds + 40
G
OUTPUT seconds
I
STOP

<table>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>10</td><td>a</td><td>i</td><td>'o' A. without quote marks</td><td>1</td></tr>
<tr><td>10</td><td>a</td><td>ii</td><td>'d' A. without quote marks</td><td>1</td></tr>
<tr><td>10</td><td>b</td><td></td><td>The complete and correct trace table is:

<table>
<tr><th>strIn</th><th>isPalindrome</th><th>iUp</th><th>iDown</th></tr>
<tr><td>abcaba</td><td>true</td><td>1</td><td>6</td></tr>
<tr><td></td><td></td><td>2</td><td>5</td></tr>
<tr><td></td><td></td><td>3</td><td>4</td></tr>
<tr><td></td><td>false</td><td>4</td><td>3</td></tr>
</table>

1 mark for isPalindrome first value true;
1 mark for isPalindrome last value false;
1 mark for iUp starting at 1;
1 mark for iUp incrementing by 1 and ending at 4;
1 mark for iDown starting at 6;
1 mark for iDown decrementing by 1 and ending at 3;

I. Anything written in the strIn column.</td><td>6</td></tr>
<tr><td>10</td><td>c</td><td>i</td><td>Syntax;</td><td>1</td></tr>
<tr><td>10</td><td>c</td><td>ii</td><td>Logical;</td><td>1</td></tr>
</table>