Candidates answer on the question paper.

Candidate
Name

Centre
Number

Candidate
Number

INSTRUCTIONS TO CANDIDATES

- Answer **all** the questions.

INFORMATION FOR CANDIDATES

- The number of marks is given in brackets [ ] at the end of each question or part of question.
- The total number of marks for this paper is **100**.

ADVICE TO CANDIDATES

- Read each question carefully and make sure you know what you have to do before starting your answer.

FOR EXAMINER'S USE		
	Max	**Mark**
1	30	
2	11	
3	37	
4	22	
TOTAL	100	

This document consists of **14** printed pages and **2** blank pages.

SP (SLM) T12103   © OCR 2007 [QAN 500/2279/9]   OCR is an exempt Charity   **[Turn Over**

Answer **all** questions.

1   An electronic general knowledge game displays, on the screen, the following:

- a question

- four possible answers

- a clock to allow a set amount of time to answer the question

- a score (which is added to if the question is answered correctly)

- the score required in order to win the game.

The player touches the answer that they want to input as their choice.

(a)   On the blank screen below, show a suitable layout to present the five areas.

[7]

The system will also keep the 100 highest scores. These are held in a file which contains, for each high score, the following:

- the name of the player

- the score they achieved

- the date on which the score was achieved

- the average time, in minutes, spent on each question.

(b) Complete the following table.

	Data type	Size of field
Name		
Score		
Date		
Time		

[8]

(c) Estimate the size, in kB, of the file of 100 records. Show your working.

....

....

....

....

....

....

....

....**[5]**

(d) The records are held in a sequential file in score order.

(i) Explain what is meant by a sequential file and give **two** reasons why it should be used in this case.

Sequential file..

....

....

....**[2]**

Reason 1 ....

....

Reason 2 ....

....**[2]**

(ii) Describe the process for adding a new high score into a file that already contains 100 high scores.

The quality of your written communication will be assessed in your answer to this question.

....

....

....

....

....

....

....

....

....

....

....

....**[6]**

2 A program is written which will use as input the marks gained by candidates in an AS examination paper. The top mark possible is 100. The program will calculate the mean (average) mark and output the highest mark, the lowest mark and the mean.

(a) It is decided to test the module using groups of four test marks at a time.

Explain why the testing was carried out using only four marks at a time.

....

....

....

....**[2]**

(b) Using the table below, give **three** separate test cases for testing the program.

Input data	Reason for test	Expected result

[9]

3 A television talent contest allows the audience to vote for one of three contestants, A, B or C, using a voting pad.

(a) Rapid Application Development (RAD) is to be used in the development of the software for controlling the voting.

(i) Explain how the use of RAD can speed up the development process.

....

....

....

....**[2]**

(ii) Explain how the end user is involved during the testing and refining of the software.

....

....

....

....**[2]**

(iii) Describe **two** advantages, for the user, of using a RAD approach.

Advantage 1 ....

....

....

....

Advantage 2 ....

....

....

....**[4]**

(b) An early version of the software assumes that there are 100 people in the audience. To store their votes, it uses VOTE_CAST, which is an array of 100 characters.

(i) Describe **one** advantage of using an array rather than 100 separate variables in this implementation.

....

....

....

....**[2]**

(ii) Give **two** items which need to be stated when defining an array, giving **one** reason for each item.

Item 1 ....

Reason 1 ....

Item 2 ....

Reason 2 ....**[4]**

(c) A second array, VOTES, contains three integers and is used with the array VOTE_CAST in the following algorithm.

```
01  FOR i = 1 TO 100
02    IF VOTE_CAST(i) = "A" THEN
03      VOTES(1) = VOTES(1) + 1
04    ELSE
05      IF VOTE_CAST(i) = "B" THEN
06        VOTES(2) = VOTES(2) + 1
07      ELSE
08        VOTES(3) = VOTES(3) + 1
09      ENDIF
10    ENDIF
11  NEXT i
12  OUTPUT A,B,C
```

(i) Explain why it will be necessary to initialise the array VOTES() before the algorithm is run.

(In this context, 'initialise' means fill the array with starter values.)

....

....

....

....**[2]**

[Turn Over

(ii) Write a FOR loop which can be used to initialise the array VOTES in the beginning of the algorithm.

....

....

....

....

....

....

....

....**[4]**

(iii) Explain what happens when the algorithm is executed.

....

....

....

....

....

....

....

....**[4]**

(iv) By stating the type of operator in each case, explain why the use of the '=' signs in lines 2 and 3 are different.

....

....

....

....

....

....**[3]**

(v) Line 10 is meant to output the total votes for A, B and C. It does not work. Rewrite line 10 to produce the correct result.

....

....

....

....**[2]**

(d) The following algorithm is written to determine which of A, B and C gets the highest vote.

```
01 IF VOTES(1) > VOTES(2) THEN
02    IF VOTES(1) > VOTES(3) THEN
03       OUTPUT "A"
04    ELSE
05       OUTPUT "C"
06    ENDIF
07 ELSE
08        IF VOTES(2) > VOTES(3)THEN
09       OUTPUT "B"
10    ELSE
11       OUTPUT "C"
12    ENDIF
13 ENDIF
```

(i) Some people do not vote and the result of a particular vote is that all of A, B and C receive equal votes.

State the line numbers that will be executed by the algorithm and which of A, B or C will be output.

....

....

....

....**[4]**

(ii) Explain how the algorithm would need to be altered to deal with two or three of the options receiving equal votes. Do **not** produce the algorithm.

....

....

....

....

....

....

....

....**[4]**

4 A telephone company is producing software to calculate the bills for its customers. A text file which contains a list of the calls for the customer is used. An extract of this file is given below.

Date	Time	Number	Duration
01/01/2007	00:01	0202332981	1:04
01/01/2007	00:23	0121928192	29:52
02/01/2007	07:45	0870736728	112:19

The company operates a flat rate so that the cost of calls depends only on the duration, and not on the destination number or the time of day.

(a) The company decides to produce the program using a top-down modular design.

(i) Explain what is meant by a top-down design.

....

....

....

....**[2]**

(ii) State **three** advantages and **one** disadvantage of dividing a problem into modules for coding.

Advantage 1 ....

....

Advantage 2 ....

....

Advantage 3 ....

....

Disadvantage ....

....**[4]**

(iii) The development team decides to divide the problem into four modules as shown in the diagram below.

Extend the diagram by dividing the module 'Get Total Duration' into further sub-modules. Do **not** attempt to divide any of the other modules further.

```
                        ┌──────────────┐
                        │ Process Bill │
                        └──────┬───────┘
        ┌──────────────┬───────┴───────┬──────────────┐
┌───────────────┐ ┌──────────────────┐ ┌────────────┐ ┌────────────────┐
│   Open File   │ │ Get Total Duration│ │ Close File │ │ Calculate Cost │
└───────────────┘ └──────────────────┘ └────────────┘ └────────────────┘
```

[4]
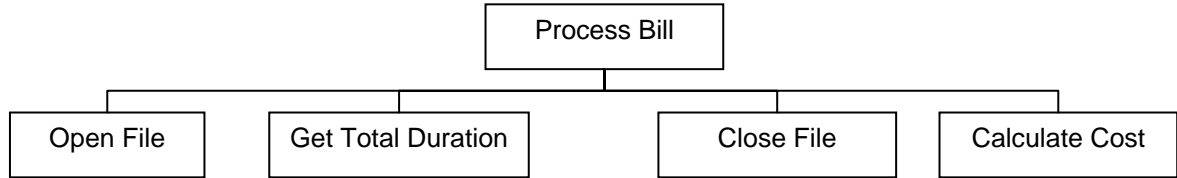
(b) (i) Functions and procedures are both examples of subprograms.

Explain how a function differs from a procedure.

....

....

....

....**[2]**

(ii) Describe **two** program writing techniques that can be used in the code of a program to facilitate ongoing maintenance.

Technique 1...

....

....

....

Technique 2...

....

....

....**[4]**
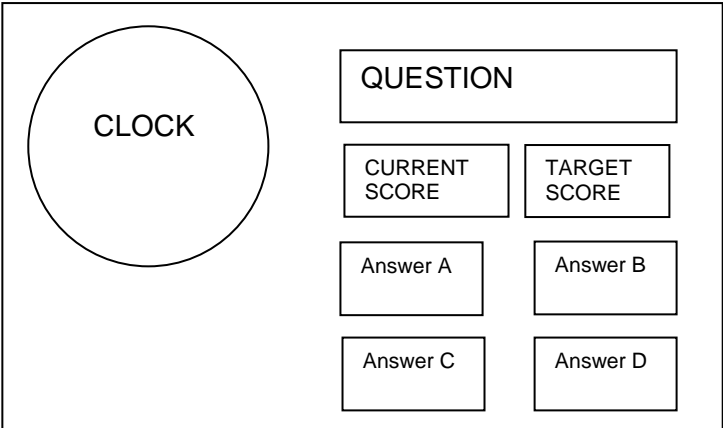
(iii) The software for the telephone company includes a function which takes the duration of a call as a **string** and returns the length of the call in minutes. For example, if the input is "1:30" the output will be 1.5.

Here is an algorithm for this function.

```
        ( BEGIN )
            │
            ▼
      / INPUT      /
     /  Duration  /
            │
            ▼
  ┌──────────────────────────┐
  │ Extract Minutes from Duration │
  └──────────────────────────┘
            │
            ▼
  ┌──────────────────────────┐
  │ Extract Seconds from Duration │
  └──────────────────────────┘
            │
            ▼
┌──────────────────────────────────┐
│ lengthOfCall = Minutes + (Seconds ÷ 60) │
└──────────────────────────────────┘
            │
            ▼
      / OUTPUT       /
     /  lengthOfCall /
            │
            ▼
        (  END  )
```

Rewrite this function in a high level language using appropriate string manipulation functions, ensuring that your code can be easily followed by another programmer. You must state the high level language that you use.

Name of language ....

Code...

....

....

....

....

....

....

....

....

....

....

....

....

....

....**[6]**

Paper Total [100]

[BLANK PAGE]

[BLANK PAGE]

Oxford Cambridge and RSA Examinations

Advanced Subsidiary GCE

COMPUTING   **F452 MS**

Unit 2: Programming Techniques and Logical Methods

Specimen Mark Scheme

The maximum mark for this paper is **100**.

Question Number	Answer	Marks
1	An electronic general knowledge game displays, on the screen, the following: • **a question** • **four possible answers** • **a clock to allow a set amount of time to answer the question** • **a score (which is added to if the question is answered correctly)** • **the score required in order to win the game.** **The player touches the answer which they want to input as their choice.**	
1(a)	**On the blank screen below, show a suitable layout to present the five areas.** • Area shown for question (1); • areas shown for all 4 answers (1); • area for clock/timer (1); • area for current score (1); • area for target score (1); • each answer area is big enough to accommodate touch (1); • no unreasonable blank areas (1). For example: 	**[7]**
1(b)	**The system will also keep the 100 highest scores. These are held in a file which contains for each high score:** • **the name of the player** • **the score they achieved** • **the date on which the score was achieved** • **the average time, in minutes, spent on each question.**	

Question Number	Answer	Marks
1(b) cont'd	**Complete the following table.**	

	Data type	**Size of field**
Name	String	10-20
Score	Integer	1/2/4/8
Date	Date (or suitable alternative but not string)	2/4/8
Time	Real	4/8

Question Number	Answer	Marks
	[1 per field]	**[8]**
1(c)	**Estimate the size, in kB, of the file of 100 records, showing your working.** *Allow follow through from 1(b).* • Record = total of field lengths from 1(b) (17 to 44) • result multiplied by 100 • divided by 1024 (or 1000) • 10% overheads added • correct answer calculated.	**[5]**
1(d)(i)	**State what is meant by a sequential file and give two reasons why it is being used in this case.** • Records are ordered logically… • according to a (key) field in the record • in this case, it allows the data to be displayed in the correct order • searched easily in the correct order.	**[4]**
1(d)(ii)	**Describe the process for adding a new high score into a file that already contains 100 high scores.** **High level response [5-6 marks]** Candidates will show a clear understanding of the question and answer the question with complete and comprehensive descriptions. The information will be presented in a structured and coherent form. There will be few if any errors in spelling, grammar and punctuation. Technical terms will be used appropriately and correctly. **Medium level response [2-4 marks]** Candidates will show an understanding of the question and answer the question with some detailed descriptions. The information will be presented in a structured format. There may be occasional errors in spelling, grammar and punctuation. Technical terms will be mainly correct.	

1 (d)(ii) cont'd	**Low level response [0-1 mark]**	
	Candidates will demonstrate a limited understanding of the question. Information may be a list of points, with little or no descriptions. Information will be poorly expressed and there will be a limited, if any, use of technical terms. Errors of grammar, punctuation and spelling may be intrusive. **Points to be made include:** • Open high score file with read access, open new file with write access • copy scores from old to new • until point of insertion • insert score to be entered into new file • copy rest of records from the old file • except the last which is discarded • delete old file/replace old file with new.	
	Alternative solution: • Load file into memory • Into an appropriate data structure (such as an array) • copy nth entry to position n+1 • repeated from position 99 • until point of insertion • insert score to be entered • save the data from data structure/array back into the file.	**[6]**

Question Number	Answer	Marks
2	A program is written which will use as input the marks gained by candidates in an AS examination paper. The top mark possible is 100. The program will calculate the mean (average) mark and output the highest mark, the lowest mark and the mean.	
2(a)	It is decided to test the module using groups of four test marks at a time. In reality, many thousands of marks will be input for each run. Explain why the testing was carried out using only four marks at a time. • Manageable number for input; • expected results can be calculated easily. [1 per bullet]	**[2]**
2(b)	Using the table below, give <u>three</u> separate test cases for testing the program.	

Input data	Reason for test	Expected result
10,20,30,40	Normal data	40,10,25
10,20,30,41	Can program handle real result?	41,10,25.25
0,20,30,100	Borderline values	100,0,37.5
0,20,30,101	Invalid values	Error message

Input data and expected results above are examples. Other values acceptable if they match the reason for test.

[3 per row, max 3 rows, max 9]

[9]

Question Number	Answer	Marks
3	A television talent contest allows the audience to vote for one of three contestants, A, B or C, using a voting pad.	
3(a)(i)	Rapid Application Development (RAD) is to be used in the development of the software for controlling the voting. Explain how the use of RAD can speed up the development process. • Prototypes of the software are produced • which have reduced functionality • and can be produced quickly • [1 per bullet, max 2]	**[2]**
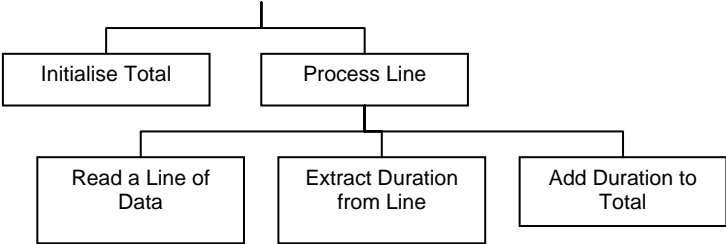
Question Number	Answer	Marks
3(a)(ii)	**Explain how the end user is involved during the testing and refining of the software.** • Prototypes are tested/evaluated with the end user; • outcome is used to inform the next prototype; • process is repeated/iterative development; • until final product is produced. [1 per bullet, max 2]	**[2]**
3(a)(iii)	**Describe two advantages, for the user, of using a RAD approach.** • End user can see a working prototype sooner/can see something happening. • End user is involved in the design/can influence the direction the program is taking. • Overall development time is shorter/reducing development costs. [2 per advantage, max 2 advantages, max 4]	**[4]**
3(b)	**An early version of the software assumes that there are 100 people in the audience. To store their votes, it uses VOTE_CAST, which is an array of 100 characters.**	
3(b)(i)	**Describe one advantage of using an array rather than 100 separate variables in this implementation.** • Code is easier to manage • …as there are fewer variables. • Can use iteration (to count up the votes)/looping • …instead of dealing with each vote separately. • Code will be more easily scaleable/they can easily change the number of voters • …by changing the size of the array. [Max 2, in pairs]	**[2]**
3(b)(ii)	**Give two items which need to be stated when defining an array, giving one reason for each item.** • Name of array… • …to allow individual data items to be accessed. • Maximum number of elements/size of array/bounds of array • …to enable contiguous locations in memory to be reserved. • Data type of contents • …to allow correct variables/to determine rules for manipulation. • Dimension • …to allow position in array to have meaning. [2 per pair, max 2 pairs, max 4]	**[4]**

Question Number	Answer	Marks

3(c)	**A second array, VOTES, contains three integers and is used with the VOTE_CAST array in the following algorithm.**
	`01  FOR i = 1 TO 100`
	`02    IF VOTE_CAST(i) = "A" THEN`
	`03      VOTES(1) = VOTES(1) + 1`
	`04    ELSE`
	`05      IF VOTE_CAST(i) = "B" THEN`
	`06        VOTES(2) = VOTES(2) + 1`
	`07      ELSE`
	`08        VOTES(3) = VOTES(3) + 1`
	`09      ENDIF`
	`10    ENDIF`
	`11  NEXT i`
	`12  OUTPUT A,B,C`	

3(c)(i)	**Explain why it will be necessary to initialise the array VOTES() before the algorithm is run. Initialise means to fill the array with starter values.** • The totals will include previous results; • the values in VOTES must be set to 0.	**[2]**
3(c)(ii)	**Write an algorithm using a FOR loop which will initialise the array VOTES.** • FOR J=1 TO 3 • VOTES(J)=0 • NEXT Mark points: • Use of FOR LOOP; • with correct condition; • attempt to set values to 0; • using correct subscript.	**[4]**
3(c)(iii)	**Explain what happens when the algorithm is executed.** • Takes each vote in turn; • decides whether it is A, B or C; • keeps a running total of the number of votes for each of A, B and C; • outputs the 0,0,0/any output given would be meaningless.	**[4]**
3(c)(iv)	**By stating the type of operator in each case, explain why the use of the '=' signs in lines 2 and 3 are different.** • In line 2, operator is relational/comparison operator; • …returns a value TRUE or FALSE. • In line 3, operator is an assignment/arithmetic operator; • …changes the value held in a variable.	**[3]**

Question Number	Answer	Marks
3(c)(v)	**Line 10 is meant to output the total votes for A, B and C. It does not work.** **Rewrite line 10 to produce the correct result.** • VOTES(1), VOTES(2), VOTES(3) Mark points: • Use of array VOTES; • all correct.	**[2]**

3(d)	**The following algorithm is written to determine which of A, B and C gets the highest vote.**
	
	<pre>01 IF VOTES(1) > VOTES(2) THEN
	02 IF VOTES(1) > VOTES(3) THEN
	03 OUTPUT "A"
	04 ELSE
	05 OUTPUT "C"
	06 ENDIF
	07 ELSE
	08 IF VOTES(2) > VOTES(3)THEN
	09 OUTPUT "B"
	10 ELSE
	11 OUTPUT "C"
	12 ENDIF
	13 ENDIF</pre>	

3(d)(i)	**Some people do not vote and the result of a particular vote is that all of A, B and C receive equal votes.** **State the line numbers that will be executed by the algorithm and which of A, B or C will be output.** • 1,7 • 8,10 • 11,12,13 • C	**[4]**
3(d)(ii)	**Explain how the algorithm would need to be altered to deal with two or three of the options receiving equal votes. Do <u>not</u> produce the algorithm.** • Check if all 3 equal; • …output suitable response/output "All equal scores"; • … check if two are equal; • …three times to cover the three possible pairs; • …IF the third vote is smaller. [1 per bullet, max 4]	**[4]**

Question Number	Answer	Marks
4(a)(i)	**A telephone company is producing software to calculate the bills for its customers. A text file which contains a list of the calls for the customer is used. An extract of this file is given below.**	

Date	Time	Number	Duration
01/01/2007	00:01	0202332981	1:04
01/01/2007	00:23	0121928192	29:52
02/01/2007	07:45	0870736728	112:19

Question Number	Answer	Marks
	The company operates a flat rate so that the cost of calls depends only on the duration, and not on the destination number or the time of day.	
4(a)	**The company decides to produce the program using a top-down modular design.**	
4(a)(i)	**State two advantages of using a top-down design.**	
	• Problem is split into smaller sub-problems • …which, in turn, are split into smaller sub-problems • …until each is one element of the algorithm. [1 per bullet; max 2]	**[2]**
4(a)(ii)	**State three advantages and one disadvantage of dividing a problem into modules for coding.**	
	Advantages • Smaller problems are easier to solve/understand • …and easier to test • …and easier to debug. • Development can be shared between a team of programmers • …according to individual strengths • use of library modules • code can be reused. [1 per advantage, max 3] Disadvantage • Modules must be linked • Programmers must ensure that cross-referencing is done • Interfaces between modules must be planned • Testing of links must be carried out [1 per disadvantage, max 1]	**[4]**

Question Number	Answer	Marks
4(a)(iii)	**The development team decides to divide the problem into four modules as shown in the diagram below.** **Extend the diagram by dividing the module 'Get Total Duration' into further sub-modules. Do not attempt to divide any of the other modules further.** • Initialise total; • read a line of data; • extract duration; • add duration to total; • attempt at a further layer; • candidate's attempt shows horizontal order. For example: 	**[4]**
4(b)(i)	**Functions and procedures are both examples of subprograms.** **Explain how a function differs from a procedure.** • A function returns a single value to the calling program so that it can be used in… • …the same way as a variable in the main body of the program • A procedure can return none or many.	**[2]**
4(b)(ii)	**Describe two program writing techniques that can be used in the code of a program to facilitate ongoing maintenance.** • Use of sensible variable names • …to allow others to understand scope of variables • …also applies to names of functions/procedures. • Indentation of code/spacing of code • …to show clearly the lines of code which should be treated together • …typically loops/selection. • Annotate code • ... with comments that are <u>not used by the computer</u> • …so that others can understand reasons for code/structures [I per technique and 1 per relevant point; max 2 techniques, max 4 marks]	**[4]**

Question Number	Answer	Marks
4(b)(iii)	The software for the telephone company includes a function which takes the duration of a call as a <u>string</u> and returns the length of the call in minutes. For example, if the input is "1:30" the output will be 1.5.	
	Here is an algorithm for this function.	
	   BEGIN INPUT Duration Extract Minutes from Duration Extract Seconds from Duration lengthOfCall = Minutes + (Seconds ÷ 60) OUTPUT lengthOfCall END **Rewrite this function in a high level language using appropriate string manipulation functions, ensuring that your code can be easily followed by another programmer. You must state the high level language that you use.** Solution (must follow the flow chart) • Function takes a string as input; • the number of minutes is correctly extracted from the input string (i.e. all the characters before the colon); • the number of seconds is correctly extracted from the input string (i.e. all the characters after the colon/last two characters); • the extracted strings are correctly converted into integers (or other numeric type); • the length of call is correctly calculated; • the length of call is returned (as a real number). [1 per bullet, max 5] Maintainable code • Appropriate identifier names used throughout; • code annotated and can be followed. [1 mark]	**[6]**

Assessment Objectives Grid (includes QWC)

Question	AO1	AO2	Total
1	11	19	**30**
2	3	8	**11**
3	14	23	**37**
4	12	10	**22**
Totals	**40**	**60**	**100**