

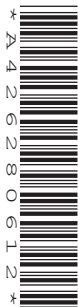
**Released June 2010  
For Assessment Submission  
January 2013 to June 2015**

**GCSE COMPUTING**

**A452**      Low Level Programming

**CONTROLLED ASSESSMENT MATERIAL 1**

This assessment may be periodically reviewed. Please check on OCR Interchange that you have the Controlled Assessment material valid for the appropriate assessment session.



**INSTRUCTIONS TO TEACHERS**

- Please refer to Section 4 of the GCSE Computing specification for instructions on completing this controlled assessment task.
- Each task can be contextualised appropriately to suit facilities available in your centre.
- The marking criteria should be available to candidates whilst completing the task.
- The quality of written communication will be assessed in the judgements and conclusion section.
- The total number of marks for this unit is **45**.

**INFORMATION FOR CANDIDATES**

- This document consists of **4** pages. Any blank pages are indicated.

**Teachers are responsible for ensuring that assessment is carried out against the Controlled Assessment set for the relevant examination series (detailed above).**

**Assessment evidence produced that does not reflect the relevant examination series will not be accepted.**

The purpose of this unit is to carry out a practical investigation of a topic chosen from a set of options supplied by OCR. In the course of the investigation, there will be an opportunity to look in depth at an aspect of computing that goes beyond the subject matter outlined in A451. The tasks will require a significant element of practical activity, which must be evidenced in the report and which will form a major element of the assessment. The topics will enable practical investigation and some supplementary research to be carried out in a variety of ways. These will include, but are not restricted to:

- practical investigations with hardware or software
- practical investigations with online resources

Supplementary research may be required and resources may include:

- web-based enquiry
- contact with IT professionals
- research using computer industry publications

**Candidates should complete all tasks.**

### Low level programming

This assignment consists of exercises using the Little Man Computer (LMC). This is a simulation of a real computer that has visible memory and a processor. You can see the registers and it is easy to keep track of what is in them throughout the running of a program.

There are various implementations of the Little Man Computer and the instructions may vary slightly between them. This assignment assumes that you are using the following example of the LMC.

<http://www.atkinson.yorku.ca/~sychen/research/LMC/LMCHome.html>

The applet itself is found at:

<http://www.atkinson.yorku.ca/~sychen/research/LMC/LittleMan.html>

Here is the instruction set for this version of LMC.

Instruction	Mnemonic	Code
LOAD	LDA	5
STORE	STA	3
ADD	ADD	1
SUBTRACT	SUB	2
INPUT	INP	901
OUTPUT	OUT	902
BRANCH IF ZERO	BRZ	7
BRANCH IF ZERO OR POSITIVE	BRP	8
BRANCH ALWAYS	BRA	6
END	HLT	000
DATA LOCATION	DAT	(the data)

If you use a different version, you may need to make slight alterations.

### Your assignment

- Here is a program written for the LMC.

```

INP
STA DEC
INP
LOOP SUB DEC
OUT
BRZ QUIT
BRA LOOP
QUIT HLT
DEC DAT

```

Enter the program and compile it.

Produce an annotated screen shot showing the results of compilation.

2. When you compile the program, values are put into some of the memory locations. Explain the logic of the program referring to these values. You may use a flow chart or any other suitable method.
3. (i) Run the program with the first input value of 2 and the second input value of 10. Explain what happens.  
(ii) Run the program again with the first input value of 3 and the second input value of 10. Explain what happens.

A problem will occur in run (ii).

- (iii) Rewrite the program to overcome this problem.
4. (i) Plan, write and test a program to run in LMC that takes two different numbers from the user and divides the larger by the smaller. It then outputs the larger number, then the smaller number and then the quotient (the whole number part only).  
(ii) Extend part (i) to calculate the whole number part of the mean of a set of numbers input by the user. The user keeps entering numbers into the computer. A value of zero indicates the end of the list of numbers. When the zero is input, the computer outputs first the total, then the mean of all the numbers input. The zero does not count as one of the numbers to be averaged.

Produce evidence to show:

- (i) that you have planned, written and tested your programs successfully
- (ii) a step by step commentary for each program explaining the purpose of each instruction.
5. Produce an evaluation of your solutions.
6. Write a conclusion about the extent that an assembly language is an efficient method for producing computer programs.



#### Copyright Information

OCR is committed to seeking permission to reproduce all third-party content that it uses in its assessment materials. OCR has attempted to identify and contact all copyright holders whose work is used in this paper. To avoid the issue of disclosure of answer-related information to candidates, all copyright acknowledgements are reproduced in the OCR Copyright Acknowledgements Booklet. This is produced for each series of examinations, is given to all schools that receive assessment material and is freely available to download from our public website ([www.ocr.org.uk](http://www.ocr.org.uk)) after the live examination series.

If OCR has unwittingly failed to correctly acknowledge or clear any third-party content in this assessment material, OCR will be happy to correct its mistake at the earliest possible opportunity.

For queries or further information please contact the Copyright Team, First Floor, 9 Hills Road, Cambridge CB2 1GE.

OCR 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.