Level 2 Creating an object oriented computer program using C++ (7540-004)

Assignment guide for Candidates
Assignment A
About City & Guilds
City & Guilds is the UK's leading provider of vocational qualifications, offering over 500 awards across a wide range of industries, and progressing from entry level to the highest levels of professional achievement. With over 8500 centres in 100 countries, City & Guilds is recognised by employers worldwide for providing qualifications that offer proof of the skills they need to get the job done.

City & Guilds Group
The City & Guilds Group includes City & Guilds, ILM (the Institute of Leadership & Management) which provides management qualifications, learning materials and membership services, NPTC which offers land-based qualifications and membership services, and HAB (the Hospitality Awarding Body). City & Guilds also manages the Engineering Council Examinations on behalf of the Engineering Council.

Equal opportunities
City & Guilds fully supports the principle of equal opportunities and we are committed to satisfying this principle in all our activities and published material. A copy of our equal opportunities policy statement is available on the City & Guilds website.

Copyright
The content of this document is, unless otherwise indicated, © The City and Guilds of London Institute 2010 and may not be copied, reproduced or distributed without prior written consent.

However, approved City & Guilds centres and learners studying for City & Guilds qualifications may photocopy this document free of charge and/or include a locked PDF version of it on centre intranets on the following conditions:

- centre staff may copy the material only for the purpose of teaching learners working towards a City & Guilds qualification, or for internal administration purposes
- learners may copy the material only for their own use when working towards a City & Guilds qualification

The Standard Copying Conditions on the City & Guilds website also apply.

Please note: National Occupational Standards are not © The City and Guilds of London Institute. Please check the conditions upon which they may be copied with the relevant Sector Skills Council.

Publications
City & Guilds publications are available on the City & Guilds website or from our Publications Sales department at the address below or by telephoning +44 (0)20 7294 2850 or faxing +44 (0)20 7294 3387.

Every effort has been made to ensure that the information contained in this publication is true and correct at the time of going to press. However, City & Guilds’ products and services are subject to continuous development and improvement and the right is reserved to change products and services from time to time. City & Guilds cannot accept liability for loss or damage arising from the use of information in this publication.

City & Guilds
1 Giltspur Street
London EC1A 9DD
T +44 (0)844 543 0000 (Centres)
T +44 (0)844 543 0033 (Learners) www.cityandguilds.com
F +44 (0)20 7294 2413 learnersupport@cityandguilds.com
Contents

Level 2 Creating an object oriented computer program using C++ (7540-004)

Assignment A

Introduction – Information for Candidates 2
Candidate instructions 3
Level 2 Creating an object oriented computer program using C++ (7540-004)

Assignment A

Introduction – Information for Candidates

About this document
This assignment comprises all of the assessment for Level 2 Creating an object oriented computer program using C++ (7540-004).

Health and safety
You are asked to consider the importance of safe working practices at all times.

You are responsible for maintaining the safety of others as well as your own. Anyone behaving in an unsafe fashion will be stopped and a suitable warning given. You will not be allowed to continue with an assignment if you compromise any of the Health and Safety requirements. This may seem rather strict but, apart from the potentially unpleasant consequences, you must acquire the habits required for the workplace.

Time allowance
The recommended time allowance for this assignment is 4 hours
Level 2 Creating an object oriented computer program using C++ (7540-004)
Candidate instructions

Candidates are advised to read all instructions carefully before starting work and to check with your assessor, if necessary, to ensure that you have fully understood what is required.

**Time allowance:** 4 hours

**Assignment set up:** A scenario is provided for candidates in the form of a company specification for new library routines.

This assignment is made up of two tasks:

- **Task A** - provides a detailed design specification that should be followed by candidates when developing their program.
- **Task B** - provides presentation criteria that should be followed by candidates when producing their work.

**Scenario**

A multinational publishing company, AZ Publishing, is developing software for validating modulus 11 numbers and dates to use as a library routine available within the company. As a contracted employee of AZ Publishing, you have been asked to create and test the software which will perform the following functions:

- Determine whether a number is a valid modulus 11 number
- Determine whether a date is valid.

The international standard for allocating reference numbers for published books (ISBN) is based on a 10 character entry. The right most character may be any number between 0 to 9 or the character X. The remaining nine characters (left most) are always numbers between the value 0 to 9. Typical examples of ISBN reference numbers are:

<table>
<thead>
<tr>
<th>ISBN</th>
<th>0-07-881442-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISBN</td>
<td>0-672-22632-4</td>
</tr>
<tr>
<td>ISBN</td>
<td>0-201-18244-0</td>
</tr>
</tbody>
</table>

ISBN references are validated using a modulus 11 check (see Appendix A). The hyphen characters are used for display purposes and are not used in the modulus 11 check and the hyphens can be stripped from the number before validation.
Task A

In this task you are required to create and test a new class called newValidation.

1 Write the code for the following methods in the newValidation class
   a public boolean validMod11(long inISBN)
      This method is passed the number as a long integer and the number is validated as a modulus 11 number as specified in Appendix A.
      This method must return true if the number is a valid modulus 11 number and false if the number is invalid.
   b public boolean validDate(int dd, int mm, int yyyy)
      This method is passed a date and must validate the date.
      This method must return true if the date is valid and false if the date is not valid.
      Note: If the year can be evenly divided by 4, but not 100, it is a leap year. However, years that are evenly divisible by 400 are also leap years.

2 Write the code to test the methods in the newValidation class.
   a The screen should be cleared and a menu should be displayed on the screen with the following options:

      MENU
      1 Validate modulus 11 number
      2 Validate date
      3 Quit
      Enter menu option:

   b If option 1 is selected the following actions should occur:
      • a prompt should be displayed requesting entry of an ISBN number
      • any hyphens in the data input should be removed and the data should be converted to a long integer and passed to the method validMod11 in the newValidation class
      • if the method validMod11 returns true then ‘Valid Modulus 11 number’ must be displayed
      • if the method validMod11 returns false then ‘Invalid Modulus 11 number’ must be displayed
      • the following prompt should be displayed ‘Press RETURN to continue’
      • when the user presses the RETURN key the screen should be cleared and the menu redisplayed.

   c If option 2 is selected the following actions should occur:
      • a prompt should be displayed requesting a date (dd-mm-yyyy)
      • the data should be passed to the validDate method in the newValidation class in the format int dd, int mm and int yyyy.
      • if the method validDate returns true then ‘Valid date’ must be displayed
      • if the method validDate returns false then ‘Invalid date’ must be displayed
      • the following prompt should be displayed ‘Press RETURN to continue’
      • When the user presses the RETURN key the screen should be cleared and the menu redisplayed.

   d If option 3 is selected the program should exit.
   e The option number must be validated in the range 1-3.
3 Prepare test data to test the newValidation class and determine the expected results.

4 Test the program, compare the expected results against the actual results and resolve any errors found. Use available debugging facilities to help locate any errors.

5 Print a listing of the code for the program.

**Task B**

*Candidates should follow the criteria below when producing their work:*

1 The program conforms to the design specification.

2 The program uses the most appropriate data type(s).

3 Meaningful names are used when declaring variables.

4 The program syntax is consistently indented to aid readability.

5 The program is commented.

**Note**

- Candidates should produce the following for their assessor:
  - A printed program listing.
  - Test data and expected results.

- At the conclusion of this assignment, hand all paperwork and removable media to the test supervisor.

- Ensure that your name is on the removable media and all documentation.

- If the assignment is taken over more than one period, all removable media and paperwork must be returned to the test supervisor at the end of each sitting.
Modulus 11 check

A modulus 11 check is carried out as follows:

Multiply each digit in the number, staring at the right, by the number 1, then 2, then 3 etc.

<table>
<thead>
<tr>
<th>Multiply by</th>
<th>10</th>
<th>9</th>
<th>8</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISBN number</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>7</td>
<td>7</td>
<td>5</td>
<td>4</td>
<td>X</td>
</tr>
<tr>
<td>Result</td>
<td>0</td>
<td>9</td>
<td>24</td>
<td>35</td>
<td>12</td>
<td>35</td>
<td>28</td>
<td>15</td>
<td>8</td>
<td>10</td>
</tr>
</tbody>
</table>

The result of each multiplication is added together.

\[0 + 9 + 24 + 35 + 12 + 35 + 28 + 15 + 8 + 10 = 176\]

The result of the addition is then divided by the modulus (11).

\[176 \text{ divided by } 11 = 16 \text{ reminder } 0\]

If the remainder from the division is 0 the number is a valid modulus 11 number otherwise the number is not a valid modulus 11 number.

The remainder is 0 so the number 013527754X is a valid modulus 11 number.