

May/June 2022

COS3711

Advanced Programming

80 Marks

Duration 2 Hours

EXAMINERS:

FIRST: DR CL PILKINGTON

SECOND: MR K HALLAND

EXTERNAL: DR L MARSHALL (UNIVERSITY OF PRETORIA)

This paper consists of 5 pages.

Instructions

1. You may type your answers in a word processor (and then print to PDF for submission) or handwrite your answers (and then scan to PDF).
2. Answer all questions. Please answer questions in order of appearance.
3. The mark for each question is given in brackets next to each question.
4. Note that no pre-processor directives are required unless specifically asked for.
5. Students must upload their answer scripts in a single PDF file (answer scripts must not be password protected or uploaded as "read only" files)
6. NO emailed scripts will be accepted.
7. Students are advised to preview submissions (answer scripts) to ensure legibility and that the correct answer script file has been uploaded.
8. Students are permitted to resubmit their answer scripts should their initial submission be unsatisfactory. However, only three submissions are allowed.
9. Incorrect file format and uncollated answer scripts will not be considered.
10. Incorrect answer scripts and/or submissions made on unofficial examinations platforms will not be marked and no opportunity will be granted for resubmission.
11. The mark awarded for an incomplete or illegible scanned submission will be the student's final mark. No opportunity for resubmission will be granted.
12. Submissions will only be accepted from registered student accounts.
13. Students who do not utilise the required invigilation or proctoring tools (IRIS) will be subjected to disciplinary processes.
14. Students suspected of dishonest conduct during the examinations will be subjected to disciplinary processes. UNISA has zero tolerance for plagiarism and/or any other forms of academic dishonesty.
15. Students are provided one hour to submit their answer scripts after the official examination time. Submissions made after the official examination time will be rejected by the examination regulations and will not be marked.
16. Students experiencing network or load shedding problems are advised to apply, together with supporting evidence, for an aegrotat within 3 days of the examination session.
17. Students experiencing technical problems should contact the SCSC on 080 000 1870 or email Examenquiries@unisa.ac.za or refer to [Get-Help](#) for the list of additional contact numbers. Communication received from your myLife account will be considered.

Remember to complete the Honesty Declaration when submitting your answers. By submitting your answers you are confirming that your submission is your own, unaided work.

Transporting cargo around the world is essential in ensuring that customers have access to the goods they need and want, and an application to manage such cargo is to be implemented.

All such items are packaged in some sort of container (which, for the purposes of this scenario, has a code represented as a string). Generally, there are two kinds of containers:

- (i) a box (with dimensions length, breadth, and height), and
- (ii) a cylinder (with dimensions height and diameter).

For transport, containers are packed onto pallets. Pallet details can be saved as a file, which can also be read back in again.

Question 1**[30 marks]**

- 1.1 Considering the scenario given above, draw a partial UML class diagram that captures the scenario. You should include the necessary classes, class attributes, and class relationships that are mentioned in the scenario. You do not have to include the Client/GUI class nor indicate constructors, access specifiers, or other methods in the classes you specify.
[You may use a software tool to create the UML class diagram.] (13)
- 1.2.1 Which design pattern would be used in the scenario presented above? (1)
- 1.2.2 Give two advantages of using design patterns when implementing solutions (2)
- 1.3 The container code takes the following format.
- Initial B or C character to indicate whether the container is a box or cylinder.
 - A 3-character serial number containing digits and upper and lowercase characters.
 - A final character that matches the initial B or C character.
- Write the regular expression (in quotes) that can be used to check that a container code meets the required criteria. An example of a valid code is B0CxB. Ensure that you use escape characters correctly. (8)
- 1.4 If an input mask were to be used for the input widget on the GUI, what string would be used to ensure valid values for the code required in 1.3? (5)
- 1.5 It has been suggested that the code variable/data member of a container class should be virtual. Comment on this suggestion. (1)

Question 2**[18 marks]**

The intention is to use reflective programming approaches.

- 2.1 Write the class definition (found in the class header file) of the main container class so that reflective programming approaches can be used. It should be possible to create such container objects using a code that has already been provided; however, a default code (D999D) should be used if no code is provided when the object is instantiated. (9)
- 2.2 The client wants to be able to add a comment to some of the container objects but does not want this to be available as a general requirement for all container objects. Considering the requirements for the cargo application so far, is it possible to add a comment like "Fragile" to a container object named `container`? If so, explain why and

[TURN OVER]

provide the code to do so; if not, explain in detail why not. Note that there are no marks allocated for answering yes or no. (3)

2.3 The following code is to be used to get property data from a container object.

```

1 QDomDocument d;
2 Container *c = new Container(); // assume a valid container
3 const QMetaObject *m = c->metaObject();
4 QDomElement basetag = d.createElement("container");
5 for (int i=1; i<m->propertyCount(); i++)
6 {
7     QMetaProperty prop = m->property(i);
8     QString propertyName = prop.name();
9     QString propertyValue = prop.read(c).toString();
10    QDomElement nametag = d.createElement(propertyName);
11    basetag.appendChild(nametag);
12    QDomElement valuetag = d.createElement(propertyValue);
13    nametag.appendChild(valuetag);
14 }

```

It is intended that the following XML structure be produced.

```

<container>
  <1st property name>first property value</1st property name>
  <2nd property name>first property value</2nd property name>
  <3rd property name>first property value</3rd property name>
</container>

```

2.3.1 There are errors in the code provided above. Rewrite the necessary lines to correct the error. You need only give the line number and the corrected code. (2)

2.3.2 Would the comment referred to in 2.2 above be included as an XML tag produced by this code? Explain clearly why you say so. Note that there are no marks for indicating yes or no. (2)

2.4 What do you understand by the term “meta-object”, and what is its major benefit? (2)

Question 3

[26 marks]

A model-view approach will be used to store the list of container codes that have been used.

3.1 What would be the differences between using a `QAbstractListModel` and a `QListWidget` to store the list of container codes? (4)

3.2 Supposing the code “B111B” needed to be added to the list of container codes instance named `model` below, write the code that would achieve this. (3)

```
QListWidget model;
```

3.3 It has been suggested that the `model` from 3.2 also be displayed using a `QTableView` on a separate tab in the application. How would you respond to this? (1)

[TURN OVER]

3.4 The intention is to provide a backup/restore facility for a `QListWidget` class such as the `model` given in 3.2. Write the class definition (which would be found in the header file) of a `QListWidget` or similar class that would make this backup/restore possible using a common design pattern. Explain clearly all code provided in this class. (4)

3.5 Suppose that there is a `WriteCodeList` class that will be used to send the code list over a network. This class should be run as a thread, and the following code is provided.

```
// pass the model (containing the code list) to the constructor
WriteCodeList *writer(new WriteCodeList(model));
QThread t(new QThread());
writer->moveToThread(t);
connect(t, SIGNAL(started()), writer, SLOT(writeList()));
t->start();
```

3.5.1 Provide the class declaration (that would be found in the header file) for the `WriteCodeList` class. (6)

3.5.2 How can you ensure that all memory allocated will be properly cleaned up after the thread has completed its work? You may describe this in detail or provide the necessary code (showing clearly where this code would be located). (6)

3.5.3 Which design pattern would this cleaning up in response the completion of a task (from 3.5.2) be an example of (as the cleanup would be implemented here)? (1)

3.6 Apart from the persistence factor, what is the major difference between using a Memento and Serialiser design pattern? (1)

Question 4

[6 marks]

The plan is to create a list of pallets. There should only be one such list. The following class is proposed as the way to implement this list (assuming that `Pallet` is a class that holds containers).

```
class ListOfPallets
{
public:
    ListOfPallets();
    static ListOfPallets *listOfPallets();
private:
    static ListOfPallets *list;
    QList<Pallet> pallets;
};
```

4.1 Which design pattern would be required here? (1)

4.2 Is this class definition being correctly implemented for this design pattern? If yes, state why you say so; if not, correct the code. (1)

[TURN OVER]

- 4.3 Write the code to demonstrate how you obtain an instance of the `ListOfPallets` in the correct implementation of this design pattern. (2)
- 4.4 Why would the function `listOfPallets()` be `static` in the correct implementation of the design pattern that should be used here? (2)