XML to GUI Transformation in Qt/C++ - Comprehensive Exam

Embedded Systems Academy

Fundamental Concepts

1. **Basic XML Parsing** Given the following XML, write Qt/C++ code to extract the window title:

```
<Window title="Main Application" width="800" height="600"/>
```

2. **QDomDocument Usage** Create a Qt application that loads and parses this XML structure:

```
<AppConfig>
  <Settings theme="dark" fontSize="12"/>
</AppConfig>
```

Widget Generation

3. **Button Creation** Parse this XML and generate QPushButton instances:

```
<Buttons>
  <Button text="OK" geometry="10,10,100,30"/>
  <Button text="Cancel" geometry="120,10,100,30"/>
</Buttons>
```

4. **Dynamic Layouts** Implement a QVBoxLayout from this XML:

```
<VerticalLayout spacing="5" margin="10">
  <Label text="Username:"/>
  <LineEdit placeholder="Enter username"/>
  <Label text="Password:"/>
  <LineEdit placeholder="Enter password" echoMode="password"/>
</VerticalLayout>
```

Advanced Widget Properties

5. **Styled Widgets** Create styled widgets from this XML:

```
<StyledWidgets>
  <Label text="Warning!" style="color:red; font-weight:bold;"/>
  <Button text="Submit" style="background-color:green;
      color:white;"/>
</StyledWidgets>
```

6. **Complex Components** Generate a QTabWidget from this structure:

```
<TabWidget>
  <Tab title="General">
    <Label text="General Settings"/>
  </Tab>
  <Tab title="Advanced">
    <CheckBox text="Enable logging"/>
  </Tab>
</TabWidget>
```

Data Binding

7. **Simple Binding** Connect an XML-defined slider to a progress bar:

```
<Bindings>
  <Slider name="volume" range="0,100" value="50"/>
  <ProgressBar name="volumeDisplay" bindTo="volume"/>
</Bindings>
```

8. **Conditional Visibility** Implement visibility rules:

```
<ConditionalWidgets>
  <Label text="Admin Options" visible="userLevel==admin"/>
  <Button text="Restart" enabled="systemStatus==idle"/>
</ConditionalWidgets>
```

Real-World Patterns

9. **Menu System** Create a complete menu from XML:

```
<MenuBar>
  <Menu title="File">
    <Action text="New" shortcut="Ctrl+N"/>
    <Action text="Open" shortcut="Ctrl+O"/>
    <Separator/>
    <Action text="Exit" shortcut="Alt+F4"/>
  </Menu>
  <Menu title="Edit">
    <Action text="Copy" shortcut="Ctrl+C"/>
  </Menu>
</MenuBar>
```

10. **Validation System** Add validation to form fields:

```
<Form>
  <Field name="email" type="email" required="true"/>
  <Field name="age" type="number" min="18" max="99"/>
</Form>
```

Performance Optimization

11. **Lazy Loading** Implement tab content loading only when tab becomes visible:

```
<TabWidget lazyLoad="true">
  <Tab title="Heavy Content">
    <!-- Content loaded only when selected -->
  </Tab>
</TabWidget>
```

12. **Widget Recycling** Modify your parser to reuse existing widgets when XML changes.

Error Handling

13. **Malformed XML** Handle these error cases:
    - Missing closing tags
    - Invalid attribute values
    - Undefined widget types
14. **Validation Schema** Implement XSD validation before parsing.

Extension Challenges

15. **Animation System** Parse and implement animations:

```
<Animation target="widget1" duration="300">
  <Property name="opacity" from="0" to="1"/>
  <Property name="geometry" from="0,0,0,0" to="100,100,200,50"/>
</Animation>
```

16. **Internationalization** Implement multilingual support:

```
<Label text="%greeting%"/>
<!-- Where %greeting% is replaced with locale-specific string -->
```

17. **Plugin Architecture** Design a system where widget handlers can be added via plugins.