National University of Computer and Emerging Sciences

Lahore Campus

Software Re-Engineering (SE4001)

Sessional-II Exam

Date: April 5, 2024

Course Instructor(s)

Dr. Farooq Ahmed

Total Time (Hrs): 1

Total Marks: 30

Total Questions: 2

_____ _____ _____

Roll No Section Student Signature

Do not write below this line

Attempt all the questions.

CLO 3: Apply different approaches to understand code

Question 1 **[10+5 marks]**

Consider the following statement based on ternary operators in C++ as an illustration:

maxNum = (A > B) ? ((A > C) ? A : C) : ((B > C) ? B : C);

(a) Write grammar rules for the general case of representing a nested ternary operator's based conditional expression / statement in C++. Nesting can happen up to any level. Please note that parentheses are not always used and shall be dealt with accordingly in the grammar.

(b) Show the parse tree of the given expression against your grammar in part (a)

CLO 5: Implement refactoring strategies for effective re-engineering

Question 2 **[15 marks]**

Consider the code given below for a Swing application that relies on a webservice supporting both XML and JSON:

```
class XmlHandler {

  private String read throws IOException(InputStream stream){
    StringBuilder content = new StringBuilder();
    BufferedReader reader = new BufferedReader(new InputStreamReader(stream));

    while( (line = reader.readLine()) != null ){
      content.append(line);
    }
    return content.toString();
  }

  public String fetchStatus(InputStream stream){
    try{
```

```
      // read stream and extract content
      String content = read(stream);

      // parse content and return status
      XmlDocument doc = new XmlDocument( content );
      return doc.getElement("status").getTextContent();
    } catch (IOException ex) {
      return "";
    }
  }
}
```

```
class JsonHandler {

  private String read throws IOException(InputStream stream){
    StringBuilder content = new StringBuilder();
    BufferedReader reader = new BufferedReader(new InputStreamReader(stream));

    while( (line = reader.readLine()) != null ){
      content.append(line);
    }
    return content.toString();
  }

  public String fetchStatus(InputStream stream){
    try{
      // read stream and extract content
      String content = read(stream);

      // parse content and return status
      JsonObject obj = new JsonObject( content );
      return obj.getString("status");
    } catch (IOException ex) {
      return "";
    }
  }
}
```

(a) Analyze the code and list the **most relevant refactorings** applicable
(b) Show **refactored code**.