

1st SIT GROUPWORK QUESTION PAPER:

Year Long 2018/2019

| | |
|-----------------------|--|
| Module Code: | CS5004NA |
| Module Title: | Emerging Programming Platforms and Technologies |
| Module Leader: | Dhruba Sen (Islington College) |

| | |
|--------------------------------------|---|
| Coursework Type: | Group |
| Coursework Weight: | This coursework accounts for 30% of your total module grades. |
| Submission Date: | Week 12 |
| When Coursework is given out: | Week 6 |
| Submission Instructions: | <p>Submit the following to Islington College RTE department before the due date:</p> <ul style="list-style-type: none"> <input type="checkbox"/> A Report in PDF format and a zip file which includes a Netbeans Project File |
| Warning: | London Metropolitan University and Islington College takes Plagiarism seriously. Offenders will be dealt with sternly. |

Plagiarism Notice

You are reminded that there exist regulations concerning plagiarism.

Extracts from University Regulations on Cheating, Plagiarism and Collusion

Section 2.3: "The following broad types of offence can be identified and are provided as indicative examples

- (i) Cheating: including copying coursework.
- (ii) Falsifying data in experimental results.
- (iii) Personation, where a substitute takes an examination or test on behalf of the candidate. Both candidate and substitute may be guilty of an offence under these Regulations.
- (iv) Bribery or attempted bribery of a person thought to have some influence on the candidate's assessment.
- (v) Collusion to present joint work as the work solely of one individual.
- (vi) Plagiarism, where the work or ideas of another are presented as the candidate's own.
- (vii) Other conduct calculated to secure an advantage on assessment.
- (viii) Assisting in any of the above.

Some notes on what this means for students:

- (i) Copying another student's work is an offence, whether from a copy on paper or from a computer file, and in whatever form the intellectual property being copied takes, including text, mathematical notation and computer programs.
- (ii) Taking extracts from published sources without attribution is an offence. To quote ideas, sometimes using extracts, is generally to be encouraged. Quoting ideas is achieved by stating an author's argument and attributing it, perhaps by quoting, immediately in the text, his or her name and year of publication, e.g. " $e = mc^2$ (Einstein 1905)". A reference section at the end of your work should then list all such references in alphabetical order of authors' surnames. (There are variations on this referencing system which your tutors may prefer you to use.) If you wish to quote a paragraph or so from published work then indent the quotation on both left and right margins, using an italic font where practicable, and introduce the quotation with an attribution.

Further information in relation to the existing London Metropolitan University regulations concerning plagiarism can be obtained from <http://www.londonmet.ac.uk/academic-regulations>

Overview

For this coursework you are required to propose and develop an Information System. The system should have a table with title “Menu Details” for storing and displaying the menu items. The menu items must contain minimum data type equivalent to the id, category, name, range level (for instance, low, medium, high) and price of item.

Additional data can be added if required.

For and Menu Information System, the equivalent data structure can be:

Dishes Number
Dish Name within each category (e.g. lamb curry, roast beef etc.)
Category (such as lamb, chicken, pork, and vegetable)
Spiciness (very hot, medium hot, mild hot)
Price

Task A – Proposal

For Task A, you are required to submit a proposal. The proposal should provide description about your system and tools and technologies used. The content requirements are:

1. Title
2. Brief information about your system
 - List of data
 - List of features
3. Justification of the tools used for the development.

Task B – Programming

1. Requirements for the Menu Information System

For Task B, you are required to use NetBeans IDE to create a Java based Menu Information System. To create the system, you need to create a project named **MenuIS** containing a class called **MenuInfo**. Since the system stores menu information, its GUI should have a table with the title “**Menu Details**” for storing and displaying the following menu items.

The requirements for the input GUI are:

1. Radio buttons for range level;
2. Check boxes or Combo box for selecting categories;
3. Text fields for all others.
4. The table should have at least 5 categories and 12 items in your demonstration.

For the project GUI, a simple menu bar is required. It should have at least the following items:

1. File – with Open File for opening an existing file and Exit for closing the system;
2. Help – for user help files.

The project should provide functionality for searching items based on price. If two or more items in the system have the same price, only the first matching item should be displayed in a JOptionPane message box. If no such dish is found in the table, then the system should display something meaningful in a JOptionPane message box. The search method should be implemented based on the **binary search** algorithm. The system GUI should have a text field for entering the search criteria (in this case, price) and a search button named "Search dishes" for searching.

You are required to implement a function for querying how many dishes are available in a category. For this function, either a Combo box or Check boxes should be implemented to provide the search criteria (e.g. beef, lamb, chicken, pork and vegetable), and a button named "item available in category" for searching. The search result should be displayed meaningfully in a JOptionPane message box, for example "There are three pork dishes: sweet and sour pork, fried pork chops and grilled pork".

Task C - Report

1. Report requirement:

For the report, you need to describe the process of developing the system:

1. Describe how Binary search algorithm was used for the search function. Use a diagram to illustrate the algorithm.
2. Describe each method you created in the MenuInfo class

Testing should be carried out and testing evidence should be provided:

1. Test your program to show it can run in NetBeans, and include a screenshot to show running result.
2. Evidence needed for: adding item details to table; searching for item in table based on price; searching for number of dishes in a category; and opening a file from menu.
3. Evidence on system validation: appropriate dialog boxes should appear when unsuitable values or no value are entered when executing system's functions.

In the conclusion section of your report, you should evaluate your work, reflect on what you have learnt from the coursework, what difficulties/problems you may have encountered and how you overcame them.

Style and quality of your report will be marked.

2. A recommended structure for your report

1. **Cover Page** {clear and concise- e.g. Task B: Technical Report, Module Code, Module Title, names of group members and their ID numbers}

2. **Contents Page** {structure of report - section numbers, heading and page numbers}
3. **Individual Tasks** {clearly list who did what – which part of the system development/testing/report was done by whom.}
4. **Introduction** {brief description of the requirements of the coursework}
5. **Body** {main part of the report, see Report requirement section.}
6. **Conclusion** {see Report requirement section}
7. **References (Bibliography)** {demonstration of your referencing skills}

Task D – Oral Presentation

Note: If student is unable to defend his/her coursework, s/he might be penalized

~ END~

Marking Scheme:

| University Grading Scheme for Undergraduate Programmes: 2018/19 | | |
|---|--------------|---------------|
| Marking criteria | Letter grade | Mark recorded |
| C1 – Work Showing Evidence: <ol style="list-style-type: none"> <i>The proposal has been professionally written in an appealing way and perfectly presented.</i> <i>Outstanding explanation of the binary search algorithm and illustrated with diagram.</i> <i>Outstanding explanation of all the methods used in the program.</i> <i>Implemented all the components mentioned in the assessments. File opened using Open file menu and help menu bar is also implemented.</i> <i>JOptionPane Message Box is implemented with appropriate message.</i> <i>Query for how many dishes are available in a category.</i> <i>Test cases is built for the inputs, execution, conditions, testing procedure and expected results that defines a single test to be executed to achieve a particular software testing objective. Also inclusion of screenshots for every test cases.</i> <i>System validation with appropriate dialog boxes appears.</i> <i>Outstanding self-evaluation of work, reflection of what they have learnt from the coursework, what problems they faced and how they overcame.</i> <i>Outstanding format of the documentation. It includes cover page, well organized content page, outstanding explanation of the individual tasks.</i> <i>Outstanding introduction, body part of the report and conclusion.</i> <i>Outstanding programming style.</i> <i>Proper Reference, Citation and Bibliography is mentioned.</i> <i>Outstanding oral presentation(viva).</i> | A+ | 95 |
| C2 – Work Showing Evidence: <ol style="list-style-type: none"> <i>The proposal has been professionally written in an appealing way and perfectly presented.</i> <i>Excellent explanation of the binary search algorithm and illustrated with diagram.</i> <i>Excellent explanation of all the methods used in the program.</i> | A | 85 |

| | | |
|---|----|----|
| <ol style="list-style-type: none"> 4. <i>Implemented all the components mentioned in the assessments. File opened using Open file menu and help menu bar is also implemented.</i> 5. <i>JOptionPane Message Box is implemented with appropriate message.</i> 6. <i>Query for how many dishes are available in a category.</i> 7. <i>Test cases is built for the inputs, execution, conditions, testing procedure and expected results that defines a single test to be executed to achieve a particular software testing objective. Also inclusion of screenshots for every test cases.</i> 8. <i>System validation with appropriate dialog boxes appears.</i> 9. <i>Excellent self-evaluation of work, reflection of what they have learnt from the coursework, what problems they faced and how they overcame.</i> 10. <i>Excellent format of the documentation. It includes cover page, well organized content page, outstanding explanation of the individual tasks.</i> 11. <i>Excellent introduction, body part of the report and conclusion.</i> 12. <i>Excellent programming style.</i> 13. <i>Proper Reference, Citation and Bibliography is mentioned.</i> 14. <i>Excellent oral presentation(viva).</i> | | |
| <p>C3 – Work Showing Evidence:</p> <ol style="list-style-type: none"> 1. <i>The proposal has been excellently written in an appealing way and perfectly presented.</i> 2. <i>Excellent explanation of the binary search algorithm and illustrated with diagram.</i> 3. <i>Excellent explanation of all the methods used in the program.</i> 4. <i>Implemented all the components mentioned in the assessments. File opened using Open file menu and help menu bar is also implemented.</i> 5. <i>JOptionPane Message Box is implemented with appropriate message.</i> 6. <i>Query for how many dishes are available in a category.</i> 7. <i>Test cases is built for the inputs, execution, conditions, testing procedure and expected results that defines a single test to be executed to achieve a particular software testing objective. Inclusion of screenshots of at least 5 test cases.</i> | A- | 75 |

| | | |
|--|----|----|
| <ul style="list-style-type: none"> 8. <i>System validation with appropriate dialog boxes appears.</i> 9. <i>Excellent self-evaluation of work, reflection of what they have learnt from the coursework, what problems they faced and how they overcame.</i> 10. <i>Excellent format of the documentation. It includes cover page, well organized content page, outstanding explanation of the individual tasks.</i> 11. <i>Excellent introduction, body part of the report and conclusion.</i> 12. <i>Excellent programming style.</i> 13. <i>Proper Reference, Citation and Bibliography is mentioned.</i> 14. <i>Excellent oral presentation(viva).</i> | | |
| <p>C4 – Work Showing Evidence:</p> <ul style="list-style-type: none"> 1. <i>The proposal has been written in a good way and presented properly.</i> 2. <i>Good explanation of the binary search algorithm and illustrated with diagram.</i> 3. <i>Good explanation of all the methods used in the program.</i> 4. <i>Implemented all the components mentioned in the assessments. File opened using Open file menu and help menu bar is also implemented.</i> 5. <i>JOptionPane Message Box is implemented with appropriate message.</i> 6. <i>Query for how many dishes are available in a category.</i> 7. <i>Test cases is built for the inputs, execution, conditions, testing procedure and expected results that defines a single test to be executed to achieve a particular software testing objective. Also inclusion of screenshots of at least 5 test cases.</i> 8. <i>System validation with appropriate dialog boxes appears.</i> 9. <i>Good self-evaluation of work, reflection of what they have learnt from the coursework, what problems they faced and how they overcame.</i> 10. <i>Good format of the documentation. It includes cover page, well organized content page, good explanation of the individual tasks.</i> 11. <i>Good introduction, body part of the report and conclusion.</i> 12. <i>Good programming style.</i> 13. <i>Reference, Citation and Bibliography is mentioned.</i> | B+ | 67 |

| | | |
|--|----|----|
| 14. Good oral presentation(viva). | | |
| C5 – Work Showing Evidence: <ol style="list-style-type: none"> 1. The proposal has been written in a good way and presented well. 2. Good explanation of the binary search algorithm and illustrated with diagram. 3. Good explanation of the methods used in the program(At least 8 methods). 4. Implemented all the components mentioned in the assessments. File opened using Open file menu and help menu bar is also implemented. 5. JOptionPane Message Box is implemented with suitable message. 6. Query for how many dishes are available in a category. 7. Test cases is built for the inputs, execution, conditions, testing procedure and expected results that defines a single test to be executed to achieve a particular software testing objective. Also inclusion of screenshots of at least 5 test cases. 8. System validation with suitable dialog boxes appears. 9. Good self-evaluation of work, reflection of what they have learnt from the coursework, what problems they faced and how they overcame. 10. Good format of the documentation. It includes cover page, well organized content page, good explanation of the individual tasks. 11. Good introduction, body part of the report and conclusion. 12. Good programming style. 13. Reference, Citation and Bibliography is mentioned. 14. Good oral presentation with few errors(viva). | B | 63 |
| C6 – Work Showing Evidence: <ol style="list-style-type: none"> 1. The proposal has been written in an satisfactory way. 2. Satisfactory explanation of the binary search algorithm and illustrated with diagram. 3. Satisfactory explanation of all the methods used in the program. 4. Implemented at least 3 components mentioned in the assessments. File opened using Open file menu and help menu bar is also implemented. 5. JOptionPane Message Box is implemented with message. | C+ | 57 |

| | | |
|---|---|----|
| <ul style="list-style-type: none"> 6. Query for how many dishes are available in a category is not mentioned. 7. Less than 5 test cases and screenshots. 8. System validation with appropriate dialog boxes appears. 9. Satisfactory self-evaluation of work, reflection of what they have learnt from the coursework, what problems they faced and how they overcame. 10. Satisfactory format of the documentation. It includes cover page, organized content page, satisfactory explanation of the individual tasks. 11. Satisfactory introduction, body part of the report and conclusion. 12. Satisfactory programming style. 13. Reference, Citation and Bibliography is mentioned. 14. Satisfactory oral presentation with some errors(viva). | | |
| <p>C7 – Work Showing Evidence:</p> <ul style="list-style-type: none"> 1. The proposal has been written in a satisfactory way. 2. Satisfactory explanation of the binary search algorithm with no diagram. 3. Satisfactory explanation of the methods used in the program. Not all the methods are explained. 4. Not all components are Implemented. File opened using Open file menu and help menu bar is also implemented. 5. JOptionPane Message Box is not implemented where needed. 6. Query for how many dishes are available in a category is missing. 7. Test cases is built for the inputs, execution, conditions, testing procedure and expected results that defines a single test to be executed to achieve a particular software testing objective. Also inclusion of screenshots for every test cases. 8. System validation with appropriate dialog boxes appears. 9. Satisfactory self-evaluation of work, reflection of what they have learnt from the coursework, what problems they faced and how they overcame. 10. No proper format of the documentation. It includes cover page, un-organized content page, not so satisfying explanation of the individual tasks. 11. Satisfactory introduction, body part of the report and conclusion. | C | 53 |

| | | |
|--|----|----|
| <p>12. Satisfactory programming style.</p> <p>13. Reference, Citation and Bibliography is mentioned.</p> <p>14. Satisfactory oral presentation with many errors.(viva).</p> | | |
| <p>C8 – Work Showing Evidence:</p> <ol style="list-style-type: none"> 1. The proposal has been written in a satisfactory way. 2. Not so satisfying explanation of the binary search algorithm and diagram is missing. 3. Explanation of all the methods used in the program are missing. At least explanation of 5 methods is required. 4. All the components mentioned in the assessments are not implemented. File to be opened using Open file menu is containing error. Help menu bar contains error. 5. JOptionPane Message Box is implemented with appropriate message. 6. Query for how many dishes are available in a category is not found. 7. Very less test cases with no evidences(screenshots) or insufficient testing carried out to valid the system. 8. Dialog boxes are missing where actually needed. Inappropriate values are allowed which is not acceptable. 9. Explanation of self-evaluation of work is average, poor reflection of what they have learnt from the coursework. 10. Poor format of the documentation. Un organized cover page, content page, doubttable explanation of the individual tasks. 11. Introduction, body part of the report and conclusion is not sufficient. 12. Poor programming style. 13. Proper Reference, Citation and Bibliography is mentioned. 14. Average oral presentation with many errors.(viva). | D+ | 47 |
| <p>C9 – Work Showing Evidence:</p> <ol style="list-style-type: none"> 1. The proposal has been written in a just acceptable way or is missing. 2. Poor explanation of the binary search algorithm and diagram is missing or explanation is missing. 3. Explanation of all the methods used in the program are missing. 4. Not all the components work well or some are found missing. | D | 43 |

| | | |
|---|----|----|
| <ol style="list-style-type: none"> 5. <i>JOptionPane Message Box is missing or used very less where required.</i> 6. <i>Query for how many dishes are available in a category is missing.</i> 7. <i>Test cases are missing or insufficient testing carried out.</i> 8. <i>System validation with appropriate dialog boxes appears.</i> 9. <i>Self-evaluation of work, reflection of what they have learnt from the coursework, what problems they faced and how they overcame are missing or written very poorly.</i> 10. <i>Very poor format of the documentation.</i> 11. <i>Poor introduction, body part of the report and conclusion.</i> 12. <i>Very poor programming style.</i> 13. <i>Reference, citation, bibliography is missing.</i> 14. <i>Very Poor oral presentation(viva).</i> | | |
| <p>C10 – Work Showing Evidence:</p> <ol style="list-style-type: none"> 1. <i>The proposal has been poorly written and poorly presented.</i> 2. <i>Binary search algorithm is poorly explained and not illustrated with the diagram.</i> 3. <i>Methods are poorly explained, not all the methods are explained well.</i> 4. <i>Not all the components are used well and some are found missing.</i> 5. <i>JOptionPane Message Box is missing or used very less where required.</i> 6. <i>Query for how many dishes are available in a category is missing.</i> 7. <i>The program has not been properly tested. Test cases are missing. No sufficient screenshots of testing carried out to validate their program runs flawlessly.</i> 8. <i>Format of the documentation is unacceptable.</i> 9. <i>Self-evaluation, reflection and what problems they faced and how they overcame has been written very poorly.</i> 10. <i>Very poor programming style.</i> 11. <i>Introduction, body part of the documentation and conclusion are poorly written.</i> 12. <i>Very less references to prove that the student work is not copied.</i> | F1 | 37 |

| | | |
|---|----|----|
| 13. <i>Very Poor in oral presentation with many errors(viva).</i> | | |
| C11 – Work Showing Evidence: <ol style="list-style-type: none"> <i>The proposal has not been submitted.</i> <i>Binary search algorithm is not explained.</i> <i>Methods description is missing or explained very poorly and student seems not sure about the function of particular methods.</i> <i>Not all the components are used well and some are found missing.</i> <i>JOptionPane Message Box is missing or used very less where required.</i> <i>Query for how many dishes are available in a category is missing.</i> <i>The program has not been properly tested. Test cases are missing. No sufficient screenshots of testing carried out to validate their program runs flawlessly.</i> <i>Format of the documentation is unacceptable.</i> <i>Self-evaluation, reflection and what problems they faced and how they overcame has been written very poorly.</i> <i>Worst programming style.</i> <i>Introduction, body part of the documentation and conclusion are poorly written or found missing.</i> <i>No references, citations, bibliography. Content found matching with other students(plagiarism).</i> <i>Absent in oral presentation(viva).</i> | F2 | 23 |
| Fail (non-submission or submission of work which cannot be given any credit (e.g., blank submission, incorrect assignment) | F3 | 0 |