1st Sit COURSEWORK Question Paper: **Summer Semester 2022**

Module Code:	**CS4051NP**
Module Title:	**Fundamentals of Computing**
Module Leader:	**Dipendra Thapa**

Coursework Type:	**Individual**
Coursework Weight:	This coursework accounts for **60%** of your total module grades.
Submission Date:	
When Coursework is given out:	
Submission Instructions:	Submit the following to Informatics College RTE department before the due date: • **Soft copy of the report** • **Zip file with source code of the program**
Warning:	London Metropolitan University and Informatics College takes Plagiarism seriously. Offenders will be dealt with sternly.

PLAGIARISM

You are reminded that there exist regulations concerning plagiarism. Extracts from these regulations are printed overleaf. Please sign below to say that you have read and understand these extracts:

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 taking unauthorised material into an examination; consulting unauthorised material outside the examination hall during the examination; obtaining an unseen examination paper in advance of the examination; copying from another examinee; using an unauthorised calculator during the examination or storing unauthorised material in the memory of a programmable calculator which is taken into the examination; 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:

1. 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.

2. 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.

For the coursework, the students are required to develop an application based on detailed guidance on given specifications. Through the coursework students should be able to:

- Develop a ***system for costume rental***
- Describe the program
- Test the program with some sample data to demonstrate its behavior
- Write a report to present the work

- Students are required to submit weekly assignment along with related research evidence

- The guideline for the Documentation/Development part is given below:

    - Components to be included are: cover page, and table of contents, figures and tables, footer.

    - The report must have an **Introduction** section with definite goals and objectives (approx. 400 words).

    - The report must have a **Discussion and Analysis** section in which students need to explain how the program was developed.

1. Scenario

A Costume Rental shop maintains information about the various available costumes in a text file. An application needs to be created which will read the text file and display all the costumes that are available for renting purpose. Then with each transaction (renting of a costume/costumes) a note/invoice should be generated for the particular client and should be written to a file. The stock of the costumes should also be updated after each transaction. For example, if the store had 10 costume dresses, then if one of those costumes is rented by a client then the number of stock should be changed to 9. In the case of returning a costume, a note/invoice should again be generated for the client returning the costume. Also, the stock is to be updated i.e. the quantity (stock) of the costume after being returned should be increased by 1.

A sample format of the text file containing the information about the costumes is as follows:

```
Cop Costume Set, Cartmax, $15, 20
```

```
Formal Suite (Black Premium), Megaplex, $14.5, 15
Fairy Costume Full Set, DollarSmart, $18, 24
```

**1st column contains the name of the costume, 2nd column contains name of brand, 3rd column contains the price for rent (5 days), 4th column contains the quantity available*

***Note: You can use your own format and add other information too*

A note/invoice should be generated for each transaction. When a client rents a costume a note/invoice should be generated which must contain the name of the costume, name of brand, name of the client, date and time of rental and the total amount to be paid for the costume. If a client decides to rent more than one costume, then the amount should be added up for all the costumes rented.

When a client returns a costume, a note/invoice should be generated again which should contain the name of the client, name of the costume, name of brand, date and time of return. The time period of the costume rental should be set to 5 days, and if a client is late for returning the costume, a fine should be applied on a daily basis which should be written to the file itself.

** Note: The format of the notes/invoices is up to you. But each file should have a unique name.*

1. **Algorithm**
   - An algorithm should be developed for the application where everything the program does should be considered. The algorithm should be described in steps, and pseudocode and flowcharts should also be included.

1. **Data Structures**

   - The programming should be done using data structures and operations in Python for input/output, character and string processing, and data storage.
   - It can use any primitive or complex data structures which might be necessary for holding the data (pairs, lists, strings, dictionaries, etc.)
   - The choice of data structures must be specified in the report.

1. **Program**
   - The program must work in a loop, displaying the available costumes and waiting for the administrator to enter the details of the clients. The program should not close unless the administrator decides to do so.
   - The program must check the input data, displaying error messages whenever unwanted data is entered, for example, if some string value is entered where a numerical value is expected.
   - The program must be implemented in a modular way with separate functions for the different operations such as input/output, reading files, generating invoices/notes, etc.

1. Description

- The program must be described in terms of its structure and behavior.
- It can be presented using text and structural charts, flowcharts, or other diagrams as needed.
    - The report must have a **Testing** section with evidence (through appropriate screen dumps) that has been carried out for the program.
    - The report must include a **Conclusion** section where they need to present their finding of the development and research (approx. 300 words).
- Demonstration of the project.
    - If any individual student is not able to justify his/her project, then the project will be kept under plagiarism.

NOTE: The project's technicality will be judged during the demo and marked accordingly.

Marking Breakdown for Coursework	
Criteria	**Total Marks**
1. Introduction	*5*
2. Algorithm	*5*
3. Pseudocode	*5*
4. Flowchart	*5*
5. Data Structures	*5*
6. Program	
a. Correctness	*10*
b. Implementation (modularity, use of functions/classes)	*10*
c. Programing Style	*10*
d. Exception Handling	*10*
e. User interface/program usability	*10*
7. Testing	*5*

8. Conclusion	*5*
9. VIVA	*10*
10. Report Structure and Formatting	*5*
Total:	**100**