Informatics and Computer Science	**Coursework Brief Proofing & Printing Confirmation Sheet**

Module Title **Programming In Java**	Module Code **20CSCI04C**
Module Leader **Assoc. Prof. Abeer Hamdy**	Semester **Two**
Proofed by **Prof. Khaled Nagaty**	

I hereby confirm that:

- This coursework brief has been proof-read (spelling and grammar) *
- This coursework brief assesses the ILOs for the module *
- This coursework brief follows the approved template *
- All questions (and sub questions) have their marks specified *

Signed (Proof Reader): _____

Signed (Module Leader)_____

Informatics and Computer Science	**20CSCI04C** **Assignment 1** **2020/ 2021**
Module Title **Programming In Java**	
Module Leader **Assoc. Prof. Abeer Hamdy**	Semester **Two**
Assessment Weight **20% of the total module mark**	Due Date **Week 6**

Instructions to Students:

1. This is a group assignment. The **group consists of 4-5 students**. Kindly follow the instructions of the TAs regarding registration of the group.

2. Submission: The submission is via the e-learning system only, by 12:00 pm on the deadline day.

3. Assessment: Assessment will be based on the class diagram and the code submitted in addition to scheduled discussions if needed.

4. Feedback: Will be given through generic feedback in labs and specific written feedback for each group three weeks after submission.

5. You can only submit your own work. Any student suspected of plagiarism will be subject to the procedures set out in the GAR.

Cairo Monorail Ticket booking System

The objective of the assignment is to implement a seat booking and ticketing system in the Cairo Monorail Transportation. Your application should contain a **minimum** of the following classes: Admin, Train (Wagon), Passenger, Route and Ticket. Admin employees login using their account and password, then they can mainly add a new train, remove a train from service, update a train stored data. An admin could also edit his own account information, like his own password. In addition, the admins can also generate a report of a number of passengers' booked, total fare collected from a certain route.

A Train added has an id, route (origin & destination) with the distances between destinations, running times and stopping stations. The ticket fare is fixed per route, each ticket represents a ticket for a specific seat number.

A Passenger can create an account for the first time accessing your system with his personal information including the username and password saved. They can then book tickets by selecting source and destination of a route, timing slot and seat number from the available seats. The passenger could book more than one ticket allowing to choose a seat number, cancel an already booked ticket, and update an already booked ticket by changing the time slot.

There are two access levels in the system:

Admin:

1) Manage admin account.
2) Add/ Remove/ Update Trains.
3) Generate Reports.

Passenger:

1) Create and manage account.
2) Book a ticket/ cancel a booking/ update booking.
3) Select/ change seat (while booking or updating the ticket)
4) Access tickets: view list of tickets booked/ print ticket.

Requirements: Analysis, Design and Classes Implementation

1. Implement the classes necessary to develop the system. Think carefully about the attributes, methods, and the **classes' relations**.
2. You have to make use of Inheritance, Interfaces and polymorphism in your design and Implementation.
3. You are expected to make use of the Java Collections by selecting from it the data structures suitable for your application.
4. Implement the main program suitable for testing the classes implemented (no graphical user interface is required in this phase).

Submission: Each group needs to submit a pdf file and their source code as follows:

1) Team members and **a description of work division among them.**
2) A General class diagram representing the System.
3) A zipped project file with the Java source code.

Marking Scheme:

Overall design	**Group mark**	**5%**
Detailed design and Implementation of each class	**Individual mark**	**10%**
Proper use of suitable collections	**Group mark**	**2.5%**
Class Implementation & Testing in main	**Group mark**	**2.5%**