



TA Name: ..... Mark: .....

## CS251 – Introduction to Software Engineering, 2021

### Each student fills this form for his program and gives it to TA

The questions to answer about each program are included in the following form. **Print and fill this form and bring with you to the discussion.**

**Student name:** Abdelrahman Sameh Mostafa **ID:** 20190293 **Group:** S10

**Which Program (Idea for task 2) did you choose?**

I have done all the assignment on my own as no one with me in the team.

**Which of the following Java / OOP features did you use in your program?**

1. How many classes did you create and their names?

5 classes (Account, SpecialAccount, Client, CommercialClient, Bank) and the main

2. How many different access specifiers did you use and their names?

2 Access specifiers (Public, Private)

3. How many Java coding style rules did you use and which ones?

9 Rules (Method and variables names, Indentation and braces, Documentation, Method specifications, Statement-comments, Loop invariants, Code organization, Return types, Public/Private access modifiers)

4. How many Javadoc tags did you use and which ones?

4 Javadoc tags (@author, @param, @override, @return)

5. Did you use inheritance? When and why?

I used inheritance twice

- SpecialAccount class (to inherit from Account class)
- CommercialClient class (to inherit from Client class)

6. Did you use method overriding? When and why?

I used to override twice

- Before toString() (to return values of the object)
- Before withdraw() in SpecialAccount (to allow over drafting for special accounts)



7. Did you use method composition? When and why?

Used composition 3 times

- ArrayList from Account (accountsList)
- ArrayList from Client (clientsList)
- setAccount() in Client ( to set the account of the client)

8. Did you use method polymorphism? When and why?

Used polymorphism once but duplicate

- toString() (to return values of the object) in classes (Account ,Client ,CommercialClient)

**Draw in the space below a simple UML class diagram that shows your main classes, their attributes and operations and their interactions with each other.**

Visual Paradigm Online Free Edition

