

Your name: Class: Student ID: No :	SE Department
---	---------------

FINAL EXAM 20202 – PART 1 (3.5 points)

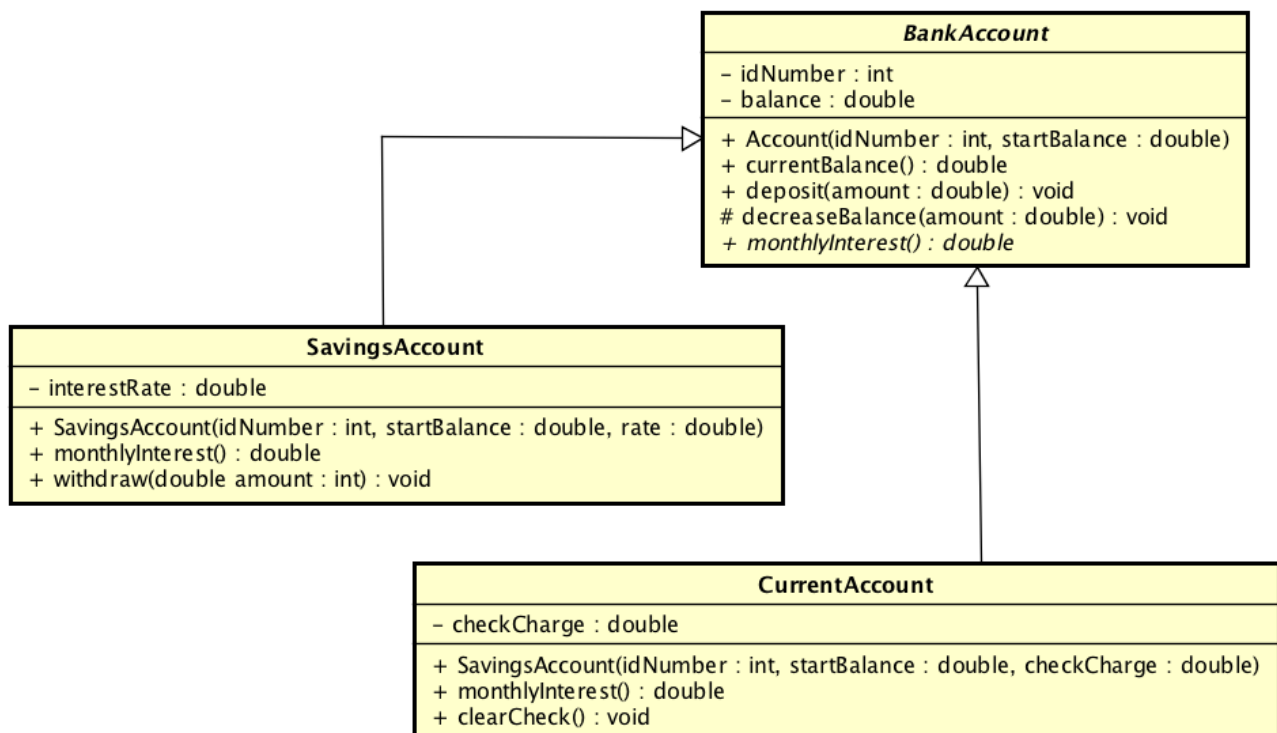
Object-Oriented Programming

Object-Oriented Language and Theory

(Duration: 30 minutes – 3.5 points – Due time: 19:00)

(All types of documents are not allowed. Any exam cheater will be given a grade of zero)

- QUESTION 1** (1.0 point – Bonus) List maximum of 2 best things you like about this course, and minimum of 3 things that should be improved.
- QUESTION 2** (1.5 points) Given a statement: “Generalization and polymorphism are core principles in object-oriented software development, which make the program more **reusable, maintainable and extensible** (i.e. easy to add new features)”. Do you agree with that statement? Why?
- QUESTION 3** (2.0 points) The class diagram below shows part of a bank account management system. Any bank account will have a given **identity** number and a **balance**. Any operation, which leads to change the balance with an invalid amount (i.e. **not positive**) or making the **balance less than zero**, will throw an **IllegalArgumentException**. The bank only pays monthly interests for **saving** accounts, but not for current accounts. The interest rate of a saving account is the rate per **year**. A current account can **clear a check** for a given amount by decreasing the balance by the amount of the check plus the per-check charge. A savings account can **withdraw** without any fee.



- a. Let x is the remainder after dividing the number comprising the last four digits in your Student ID by 2. Please write source code for classes depending on x :
- $x = 0$: **BankAccount**, **SavingsAccount** and a class with a main method to create and work

with corresponding account objects.

- $x = 1$: BankAccount, CurrentAccount and a class with a main method to create and work with corresponding account objects.

b. What is the problem with the above design? If any, please give a better solution.

Hints: Please think about object-oriented techniques for designing and implementing the problem, e.g. method overloading, inheritance, overriding, runtime polymorphism.

----- THE END -----

LƯU Ý:

1. Thời gian làm bài:

- Đề thi gồm 2 phần, Phần 1 làm trong 30 phút, Phần 2 làm trong 60 phút. Mỗi phần có 5 phút nộp bài, và thêm 10 phút nộp muộn.
- Phần 1 sẽ mở sớm 5 phút để các em đọc đề, tức là từ 18:25 (các em có lợi 5 phút đó). Phần 1 hết giờ lúc 19:05, đóng lúc 19:15 (không được nộp nữa).
- Phần 2 sẽ tính tiếp nối từ 19:05 nhưng sẽ được mở sớm từ 18:45. Phần 2 hết giờ lúc 20:10, đóng lúc 20:20.
- Nộp muộn sẽ bị trừ điểm (tối đa được nộp muộn 10 phút).

2. Cách làm bài và nộp bài:

- Viết bài làm ra giấy thi, ghi rõ họ tên, MSSV, lớp trên tất cả các mặt giấy
- Chụp ảnh/scan bài một cách rõ nét ngay khi hết giờ (trong vòng 5 phút, nộp muộn sẽ bị trừ điểm)
- Nộp file pdf hoặc file word, trong đó chứa toàn bộ các hình ảnh bài thi theo đúng thứ tự. Có thể nộp thêm file ảnh rời backup trong trường hợp file chính bị lỗi/mờ
- Cách thức đặt tên file chính (pdf/word) như sau: StudentID_FullName_Part_1, StudentID_FullName_Part_2. Nếu nộp thêm các file rời thì thêm số thứ tự phía sau, VD: StudentID_FullName_Part_1-1, StudentID_FullName_Part_1-2...
- Trong trường hợp bất khả kháng, nếu sinh viên không nộp được 1 file pdf/word, có thể nộp file rời nhưng chú ý cách đặt tên và đánh số thứ tự (không ưu tiên cách này).
- Trong lúc nộp bài (tối đa 5 phút), nếu sinh viên tắt camera điện thoại để chụp ảnh thì cần bật camera máy tính lên.