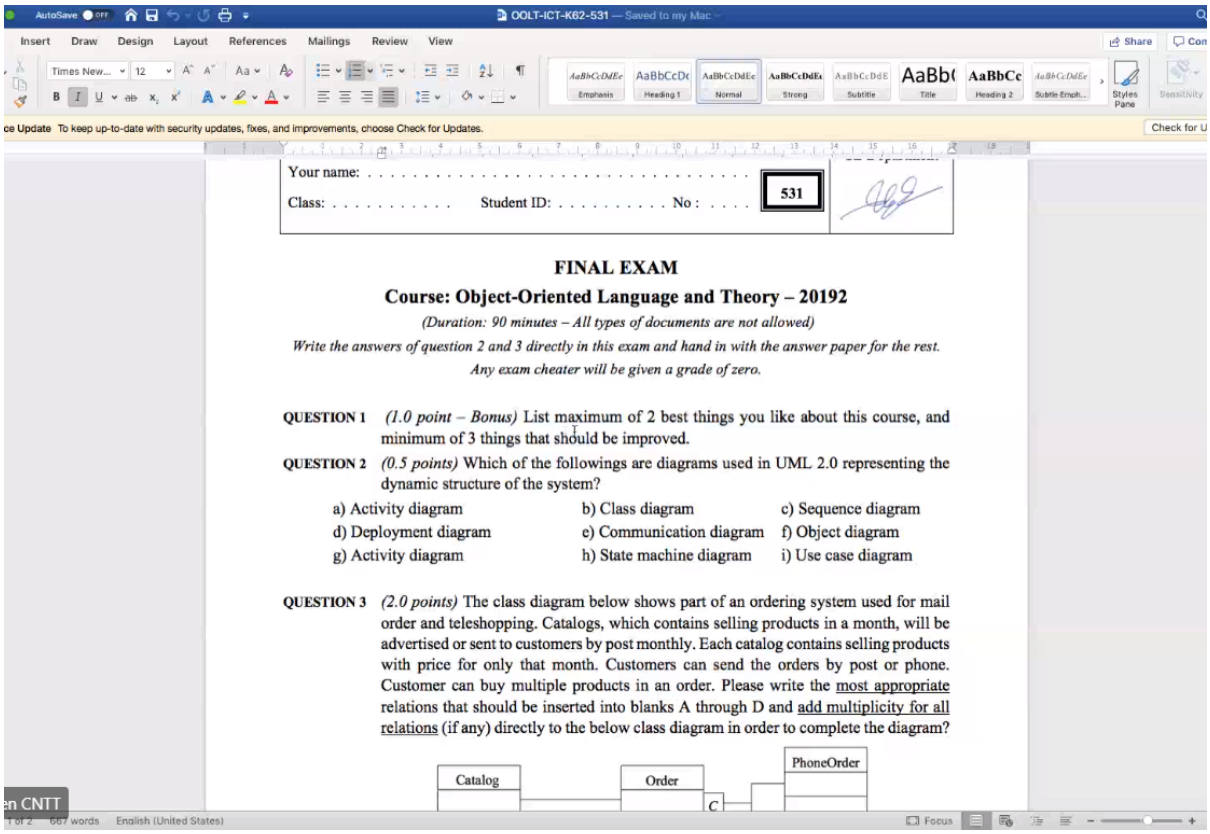
~ đề thi năm trước ko có ý nghĩa gì đâu ~

Đề K63?



Your name: 531

Class: . . . . . Student ID: . . . . . No : . . . .

FINAL EXAM

Course: Object-Oriented Language and Theory – 20192

(Duration: 90 minutes – All types of documents are not allowed)

Write the answers of question 2 and 3 directly in this exam and hand in with the answer paper for the rest.

Any exam cheater will be given a grade of zero.

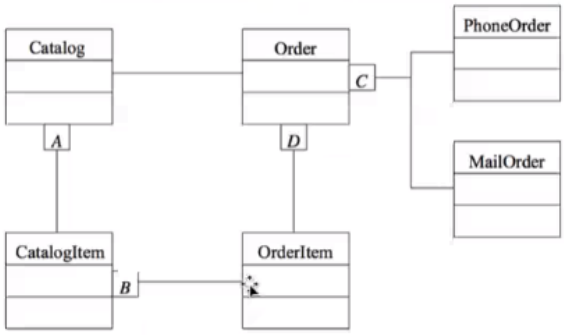
QUESTION 1 *(1.0 point – Bonus)* List maximum of 2 best things you like about this course, and minimum of 3 things that should be improved.

QUESTION 2 *(0.5 points)* Which of the followings are diagrams used in UML 2.0 representing the dynamic structure of the system?

a) Activity diagram	b) Class diagram	c) Sequence diagram
d) Deployment diagram	e) Communication diagram	f) Object diagram
g) Activity diagram	h) State machine diagram	i) Use case diagram

QUESTION 3 *(2.0 points)* The class diagram below shows part of an ordering system used for mail order and teleshopping. Catalogs, which contains selling products in a month, will be advertised or sent to customers by post monthly. Each catalog contains selling products with price for only that month. Customers can send the orders by post or phone. Customer can buy multiple products in an order. Please write the most appropriate relations that should be inserted into blanks A through D and add multiplicity for all relations (if any) directly to the below class diagram in order to complete the diagram?

QUESTION 3 *(2.0 points)* The class diagram below shows part of an ordering system used for mail order and teleshopping. Catalogs, which contains selling products in a month, will be advertised or sent to customers by post monthly. Each catalog contains selling products with price for only that month. Customers can send the orders by post or phone. Customer can buy multiple products in an order. Please write the most appropriate relations that should be inserted into blanks A through D and add multiplicity for all relations (if any) directly to the below class diagram in order to complete the diagram?



Hint: Some options could be considered to insert into the blanks A through D: Aggregation, Generalization, Composition, Navigability, Association…

A. ..... B. .....

C. ..... D. .....

Page 1/2 – Test no. 531

Mất mẹ câu 4, fml

QUESTION 5 *(6.5 points)* You are asked to develop a Java program manipulating to math expressions. An expression grammar can be defined in BNF (Backus normal form) as follows:

```
expression ::= variable | sequence
sequence   ::= expression + expression |
               expression - expression  |
               expression * expression |
               expression / expression
```

An expression can be a variable or a sequence expression.

A variable one contains only a double value while the sequence one contains two expressions and their operator (which can be plus +, minus -, multiply *, or divide /).

Any expression should be evaluated to get it's value:

- the value of a variable expression
- or the value of the operation between two values of the two expressions composing the sequence expression

An expresion can be compared to be "less than", "equal to" or "greater than" another one. It also can be tested for the equality to another expression.

In any case, if any violation to above description occurs during create new or manipulate to an expression, an **`ExpressionException`** is thrown (assume it is a user-defined exception which is a child of Exception class in Java).

a. Please design with a class diagram for the above problem and implement it in a Java program. Be careful to provide accounts with appropriate getter/setter methods and necessary constructors.

b. Write a simple main program to create different expressions, and then do some manipulations to them. Remeber to display expression information and their values after each manipulation.

sequence expression

An expresion can be compared to be "less than", "equal to" or "greater than" another one. It also can be tested for the equality to another expression.

In any case, if any violation to above description occurs during create new or manipulate to an expression, an **`ExpressionException`** is thrown (assume it is a user-defined exception which is a child of Exception class in Java).

a. Please design with a class diagram for the above problem and implement it in a Java program. Be careful to provide accounts with appropriate getter/setter methods and necessary constructors.

b. Write a simple main program to create different expressions, and then do some manipulations to them. Remeber to display expression information and their values after each manipulation.

c. Please provide the "update piece" of the class diagram if we accept new operators: a summation operator: e.g. $\sum_{i=1}^{10}(3 * i)$, an exponentiation operator: e.g. a^n, and an unary negation -b. Explain about the update for each new operator.

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

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

Đề K62?

Course: Object-Oriented Language and Theory – 20182

(Duration: 90 minutes – All types of documents are not allowed)

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)* What is encapsulation and information hiding? When should and shouldn't you use getters and setters for private attributes?

QUESTION 3 *(2.0 points)* What are abstract classes in Object-Oriented Methodology (in general, not specific in Java)? What are main purposes of abstract classes? Compare to interfaces. Give examples with runtime polymorphism.

QUESTION 4 *(6.5 points)* You are asked to develop a Java program demonstrating the example of bank operations. Customers can have multiple bank accounts. Each account has an identity number and balance. The identity number cannot be reset after the creation of the account. The balance can be changed only through transactions.

With a normal account, customers have to pay a fee of 3.000 VNĐ for each time withdrawing money to decrease a specified positive amount to its balance. With a VIP account, customers don't need to pay this fee and they can receive monthly interest at the rate of 2%/year. Customers can also register a saving account with a higher monthly interest rate (e.g. 6%/year but it can change daily). However, with this type of account, customers need to pay a higher fee when withdrawing money, i.e. 0.05% the withdraw amount (the minimum fee is 20.000 VNĐ in any case).

In any transaction or at the initial state, the balance of a normal account or a saving account must be greater than or equal to 100.000 VNĐ, while the one of a VIP account must be at least 500.000 VNĐ. The monthly fee of a normal account is 10.000 VNĐ, the one of a saving account is 20.000 VNĐ and the one of a VIP account is 50.000 VNĐ. Customers can deposit a positive amount into any account, hence increasing its balance.

In any case, if the transaction violates all above rules for bank accounts, an IllegalArgumentException is thrown.

a. *Please design with a class diagram for the above problem and implement*

Give examples with runtime polymorphism.

QUESTION 4 *(6.5 points)* You are asked to develop a Java program demonstrating the example of bank operations. Customers can have multiple bank accounts. Each account has an identity number and balance. The identity number cannot be reset after the creation of the account. The balance can be changed only through transactions.

With a normal account, customers have to pay a fee of 3.000 VNĐ for each time withdrawing money to decrease a specified positive amount to its balance. With a VIP account, customers don't need to pay this fee and they can receive monthly interest at the rate of 2%/year. Customers can also register a saving account with a higher monthly interest rate (e.g. 6%/year but it can change daily). However, with this type of account, customers need to pay a higher fee when withdrawing money, i.e. 0.05% the withdraw amount (the minimum fee is 20.000 VNĐ in any case).

In any transaction or at the initial state, the balance of a normal account or a saving account must be greater than or equal to 100.000 VNĐ, while the one of a VIP account must be at least 500.000 VNĐ. The monthly fee of a normal account is 10.000 VNĐ, the one of a saving account is 20.000 VNĐ and the one of a VIP account is 50.000 VNĐ. Customers can deposit a positive amount into any account, hence increasing its balance.

In any case, if the transaction violates all above rules for bank accounts, an IllegalArgumentException is thrown.

a. *Please design with a class diagram for the above problem and implement it in a Java program. Be careful to provide accounts with appropriate getter/setter methods and necessary constructors.*
b. *Write a simple main program to create different bank accounts with a given identity number and initial balance, and then let them do some above basic bank operations such as deposit, withdraw, get monthly interest. Remeber to display their balance after each transaction.*

Hints. *Please think about object-oriented techniques for designing and implementing*

With a normal account, customers have to pay a fee of 3.000 VNĐ for each time withdrawing money to decrease a specified positive amount to its balance. With a VIP account, customers don't need to pay this fee and they can receive monthly interest at the rate of 2%/year. Customers can also register a saving account with a higher monthly interest rate (e.g. 6%/year but it can change daily). However, with this type of account, customers need to pay a higher fee when withdrawing money, i.e. 0.05% the withdraw amount (the minimum fee is 20.000 VNĐ in any case).

In any transaction or at the initial state, the balance of a normal account or a saving account must be greater than or equal to 100.000 VNĐ, while the one of a VIP account must be at least 500.000 VNĐ. The monthly fee of a normal account is 10.000 VNĐ, the one of a saving account is 20.000 VNĐ and the one of a VIP account is 50.000 VNĐ. Customers can deposit a positive amount into any account, hence increasing its balance.

In any case, if the transaction violates all above rules for bank accounts, an IllegalArgumentException is thrown.

a. *Please design with a class diagram for the above problem and implement it in a Java program. Be careful to provide accounts with appropriate getter/setter methods and necessary constructors.*
b. *Write a simple main program to create different bank accounts with a given identity number and initial balance, and then let them do some above basic bank operations such as deposit, withdraw, get monthly interest. Remeber to display their balance after each transaction.*

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

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