

A polymorphism-based C++ application (to be solved in class):

Implement a class hierarchy for a bank Account. Name your base class as *Account* and have two derived classes called *CheckingAccount* and *SavingsAccount*. Customers can deposit and withdraw money from their accounts. The savings account earn 10% interest, checking account earns none. The accounts will have methods *getBalance*, *calculateInterest*, *credit* and *debit*. Checking account has a fee per transaction. When accounts are created you shall be specifying the initial amount, transaction fee and interest rate whenever applicable. If no value is given use the bank defaults of your choice. When asked, print out the name, account number, balance and the interest rate of the account.

Write a program that creates several checking and savings account and test their member functions. Design a menu that enables a bank teller to do transactions specified by the customer. As you process each account determine its type and calculate the amount of interest owed to the account if the account is savings account.

In your program you are expected to use all of the following:

- 1) Class hierarchy
- 2) Inheritance
- 3) Abstract classes
- 4) Polymorphism
- 5) Default parameters
- 6) Overloading
- 7) Private/Protected data members and public set/get methods

Question_2 (to be solved in class):

You are asked to calculate the cost of a computer. A computer consists of parts and each part has a price. You will be implementing a base abstract *Parts* class and several concrete part subclasses (i.e. motherboard, power supply memory, hard disk etc.). Each concrete part will have a method that tells its cost. Write a program that gives the cost of a computer. Following is an incomplete class declaration that may give you a starting point. Add whatever you want to this and *Parts* class.

```
class Computer {
public:
    void add(Parts *p);
    double price();
private:
    Parts *computer_parts[200]; // waste of space. Better way is using dynamic memory alloc.
};
```

Homework_ Question (to be solved at home):

Implement a base *TextDocument* and two derived classes *XMLDocument* and *HTMLDocument*. A document has a *text* (string) data member and two methods *print()* and *type()*. An XML document starts with the word "XML:." And HTML document starts with

“HTML:”. `type()` returns the type of the document and `print()` displays the document. Create an array of `TextDocument*` with a mixture of documents. Then write a function that lists the types of the documents. Implement *main()* to test your classes.