

Вариант 16 Банковские операции

Разработать приложение, позволяющее организовать учёт денег, хранящихся в банке. Банковские операции поддерживаются для трёх типов вкладов: обычных и срочных рублёвых, а также валютных. Операции для обычных вкладов могут выполняться с частью вклада, а для срочных – только со всем вкладом; операции с валютой аналогичны операциям с обычными рублёвыми вкладами. Каждый вклад представлен своим описателем.

Описатель обычного вклада содержит: сумму вклада, дату открытия вклада и дату последней операции с вкладом; процент начисления по вкладу (ежемесячный, т.е. начисления на вклад выполняются, начиная от даты последнего изменения по текущую дату, которая передаётся в метод).

Описатель срочного вклада содержит: сумму вклада, дату открытия вклада и срок хранения, процент начисления по вкладу (ежемесячный).

Описатель валютного вклада содержит: название валюты, сумму вклада, дату открытия вклада и дату последней операции с вкладом, курс обмена, процент начисления по вкладу (ежемесячный, начиная от даты последней операции).

Каждый вклад имеет уникальную характеристику – номер счёта. Информация обо всех вкладах сведена в таблицу, каждый элемент которой состоит из номера счёта и указателя на описатель вклада. Элементы таблицы упорядочены по возрастанию номеров вкладов.

Обеспечить выполнение следующих операций.

- ❖ Для таблицы:
 - включить новый элемент в таблицу, не нарушая упорядоченности;
 - найти элемент в таблице по номеру счёта;
 - исключить элемент из таблицы по номеру счёта;
 - показать содержимое таблицы.
- ❖ Для любого вклада:
 - получить (вернуть в качестве результата) сумму вклада; изменить сумму вклада;
 - рассчитать величину начислений по процентам;
 - получить дату открытия вклада;
 - получить дату последней операции; изменить дату последней операции;
 - показать всю информацию о вкладе.
- ❖ Для валютного вклада:
 - выполнить пересчёт суммы вклада в рублях.
- ❖ Для приложения:
 - открыть новый счёт;
 - показать текущее состояние счёта (с начислением процентов по состоянию на текущую дату);
 - закрыть счёт (с начислением процентов по состоянию на текущую дату);
 - модифицировать состояние счёта (внести деньги, снять вклад, снять проценты);
 - показать информацию обо всех имеющихся счётах
 - (*) выполнить пересчёт вкладов, хранящихся на всех счетах, в указанную валюту, используя класс-итератор.

Порядок выполнения работы

1. На основе описания задачи определить состав классов.
 2. Разработать иерархию классов и схему их взаимодействия.
 3. Для каждого класса определить его состояние и необходимые методы.
 4. Разработать и отладить все классы. Приложение реализовать в виде простой диалоговой программы.
 5. (*) Разработать и отладить класс, реализующий работу приложения. Предусмотреть в классе возможность сохранения таблицы в файле и ее восстановления.
 6. (*) Разработать и отладить класс-итератор, с помощью которого реализовать дополнительные операции.
 7. (*) Повторить разработку контейнерного класса (вместе с необходимыми итераторами), используя стандартную библиотеку STL. Выбор шаблона классов согласовать с преподавателем.
 8. (*) Повторить разработку контейнерного класса (вместе с необходимыми итераторами), используя собственные шаблоны классов.
 9. (*) Реализовать приложение с использованием средств оконного интерфейса.
- Примечание: пункты задания, отмеченные (*), являются дополнительными.