

## Вариант 14 Жилищный фонд

Разработать приложение, позволяющее организовать работу по учёту жилищного фонда некоторой территории. Информация о жилье разного типа (типовая однокомнатная квартира, комфортабельная квартира – апартаменты, коттедж) хранится в описателе жилья.

Описатель *квартиры* содержит следующую информацию: адрес квартиры (название улицы, номер дома, номер квартиры) и массив из четырех описателей *помещений* квартиры (комната, кухня, санузел, прихожая).

Описатель *апартаментов* содержит следующую информацию: адрес квартиры (название улицы, номер дома, номер квартиры); количество помещений квартиры и указатель на динамический массив описателей *помещений* квартиры.

Описатель *коттеджа* содержит следующую информацию: адрес коттеджа (название улицы, номер дома); количество строений и указатель на динамический массив описателей *строений*.

Описатель *помещения* – название помещения, площадь помещения, комментарий. Описание *строения* – номер строения, количество помещений и указатель на динамический массив *описателей* помещений.

Информация обо всём жилье сведена в просматриваемую таблицу, каждый элемент которой содержит признак состояния (заселено или нет), стоимость 1 кв. м. жилья и указатель на описатель жилья.

Обеспечить выполнение следующих операций:

- ❖ Для таблицы:
  - включить новый элемент в таблицу;
  - найти элемент по заданному адресу жилья;
  - показать содержимое таблицы.
- ❖ Для любого жилья:
  - вывести информацию о жилье;
  - получить (вернуть в качестве результата) тип жилья;
  - получить значение состояния жилья; получить стоимость 1 кв. м.; получить адрес жилья;
  - рассчитать общую площадь квартиры, строения или дома.
- ❖ Для любого строения:
  - получить информацию о помещениях строения;
  - получить значение площади строения.
- ❖ Для любого помещения:
  - вывести информацию о помещении в квартире или строении;
  - получить значение площади помещения.
- ❖ Для приложения:
  - зарегистрировать заселение в жильё (для нового адреса – создать новый элемент; для незаселённого жилья – оформить заселение);
  - зарегистрировать освобождение жилья;
  - показать информацию обо всём предложении жилья (незаселённое жильё);
  - (\*) показать информацию обо всём жилье с расчетом его стоимости, 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.