

Задание 2 для первого семестра 2 курса

Требуется разработать и реализовать параметризованный класс списочного типа. Класс должен содержать конструктор без параметров, конструктор копирования, другие конструкторы по смыслу решаемых задач.

Набор методов класса должен соответствовать идеологии работы с соответствующей схемой хранения данных, т.е. обеспечивать требуемую дисциплину доступа к данным (возможно, с некоторыми модификациями) и также иметь итератор как средство перебора элементов структуры и идентификации положения отдельных элементов.

Итераторы реализуются двумя способами (в разных вариантах заданий) — как внутренние классы и как внешние по отношению к основному классу. Должен быть реализован по крайней мере один тип итератора — неконстантный. Другие типы (обратный, константный и т.д.) — по мере необходимости.

Для класса должен быть переопределен оператор вывода <<, который в наглядной форме изображает состояние данного класса.

Для контроля за некорректными ситуациями следует использовать механизм исключений и в некоторых случаях логические возвращаемые значения.

Для класса должен быть реализован оператор присваивания и также переопределены некоторые операции, например, сложение как объединение множеств элементов.

Как правило, для класса надо будет реализовать процедуры сортировки содержимого по некоторым заданным критериям и поиска требуемого элемента (возвращается итератор на найденный элемент).

Релизация тестируется на нескольких различных типах хранимых объектов — “простых” и “сложных”. Простые объекты — это примитивные числовые типы. Более сложные — структурные типы вроде вектора или числовых классов из первого задания. Наконец, реализация должна работать для “рекурсивного” типа, т.е. стек стеков целых чисел, список списков векторов и т.п. (`Stack<Stack<int>>`).

Требования к реализации конкретного задания обговариваются в рабочем порядке.

Варианты структур данных

A. Динамический массив. Сортировка, поиск.

B1. Однонаправленный список. Внешний итератор.

B2. Однонаправленный список. Внутренний итератор.

C1. Двухнаправленный список. Внешний итератор.

C2. Двухнаправленный список. Внутренний итератор.

D1. Кольцевой однонаправленный список. Внешний итератор.

D2. Кольцевой однонаправленный список. Внутренний итератор.

E1. Кольцевой двухнаправленный список. Внешний итератор.

E2. Кольцевой двухнаправленный список. Внутренний итератор.

F1. Очередь на базе списка со вставкой/удалением в середине. Внешний итератор.

F2. Очередь на базе списка со вставкой/удалением в середине. Внутренний итератор.

G1. Дек на базе кольцевого списка. Итератор перемещения “окна”(внешний).

G2. Дек на базе кольцевого списка. Итератор перемещения “окна”(внутренний).

Типы объектов

Работа над заданием предполагает еженедельный контроль по следующему графику (естественно, его можно выполнять с опережением)

График выполнения задания

неделя 1: разработка и согласование описания (интерфейса) класса.

неделя 2: реализация некоторых методов и первичный тест на простых типах данных, проверка работы механизма исключений

неделя 3. реализация более полного набора методов, тест совместной работы этих методов.

неделя 4: тест на сложных типах данных

недели 5: доработка, исправление и окончательная сдача задания

в абсолютных датах контрольный срок сдачи задания — начало ноября.