**Подгруппа 1**

1. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их возрастания. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
2. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их убывания. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
3. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
4. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке, обратном их добавлению в контейнер. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
5. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их возрастания, при этом игнорировались бы повторяющиеся значения, то есть если следующий элемент массива равен предыдущему, то итератор возвращает не его, а пытается найти следующий по значению элемент. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
6. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их убывания, при этом игнорировались бы повторяющиеся значения, то есть если следующий элемент массива равен предыдущему, то итератор возвращает не его, а пытается найти следующий по значению элемент. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
7. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их возрастания, при этом игнорировались бы повторяющиеся значения, если их количество больше 2, то есть если следующий элемент массива равен предыдущему и таковых элементов больше 2, то итератор возвращает не его, а пытается найти следующий по значению элемент. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
8. Описать шаблон класса для хранения данных (контейнер). Данные должны храниться, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их убывания, при этом игнорировались бы повторяющиеся значения, если их количество больше 2, то есть если следующий элемент массива равен предыдущему и таковых элементов больше 2, то итератор возвращает не его, а пытается найти следующий по значению элемент. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
9. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их возрастания, при этом игнорировались бы повторяющиеся значения, если их количество больше 3, то есть если следующий элемент массива равен предыдущему и таковых элементов больше 3, то итератор возвращает не его, а пытается найти следующий по значению элемент. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
10. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их убывания, при этом игнорировались бы повторяющиеся значения, если их количество больше 3, то есть если следующий элемент массива равен предыдущему и таковых элементов больше 3, то итератор возвращает не его, а пытается найти следующий по значению элемент. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
11. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в следующем порядке: сначала максимальный элемент, затем минимальный, затем второй максимальный, затем второй минимальный и т.д. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
12. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в следующем порядке: сначала минимальный элемент, затем максимальный, затем второй минимальный, затем второй максимальный и т.д. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
13. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый четвёртый элемент, начиная с нулевого, то есть возвращал бы только элементы с индексами, делящимися на 4. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
14. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый второй элемент, начиная с нулевого, то есть возвращал бы только элементы с чётными индексами. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
15. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый второй элемент, начиная с первого, то есть возвращал бы только элементы с нечётными индексами. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.

**Подгруппа 2**

1. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в следующем порядке: сначала первый добавленный в массив элемент, затем последний, затем второй добавленный в массив элемент, затем предпоследний и т.д. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
2. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в следующем порядке: сначала последний добавленный в массив элемент, затем первый, затем предпоследний добавленный в массив элемент, затем второй и т.д. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
3. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в следующем порядке: сначала элемент, находящийся в середине массива, затем элемент справа от него, затем слева от него, затем элемент, находящийся на втором месте справа от среднего, затем – на втором месте слева от среднего и т.д. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
4. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их возрастания. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
5. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их убывания. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
6. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
7. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке, обратном их добавлению в контейнер. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
8. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их возрастания, при этом игнорировались бы повторяющиеся значения, то есть если следующий элемент массива равен предыдущему, то итератор возвращает не его, а пытается найти следующий по значению элемент. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
9. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их убывания, при этом игнорировались бы повторяющиеся значения, то есть если следующий элемент массива равен предыдущему, то итератор возвращает не его, а пытается найти следующий по значению элемент. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
10. Описать шаблон класса для хранения данных (контейнер). Данные должны храниться, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их возрастания, при этом игнорировались бы повторяющиеся значения, если их количество больше 2, то есть если следующий элемент массива равен предыдущему и таковых элементов больше 2, то итератор возвращает не его, а пытается найти следующий по значению элемент. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
11. Описать шаблон класса для хранения данных (контейнер). Данные должны храниться, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их убывания, при этом игнорировались бы повторяющиеся значения, если их количество больше 2, то есть если следующий элемент массива равен предыдущему и таковых элементов больше 2, то итератор возвращает не его, а пытается найти следующий по значению элемент. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
12. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их возрастания, при этом игнорировались бы повторяющиеся значения, если их количество больше 3, то есть если следующий элемент массива равен предыдущему и таковых элементов больше 3, то итератор возвращает не его, а пытается найти следующий по значению элемент. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
13. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый четвёртый элемент, начиная с нулевого, то есть возвращал бы только элементы с индексами, делящимися на 4. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
14. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их убывания, при этом игнорировались бы повторяющиеся значения, если их количество больше 3, то есть если следующий элемент массива равен предыдущему и таковых элементов больше 3, то итератор возвращает не его, а пытается найти следующий по значению элемент. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.

**Подгруппа 3**

1. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их убывания, при этом игнорировались бы повторяющиеся значения, если их количество больше 3, то есть если следующий элемент массива равен предыдущему и таковых элементов больше 3, то итератор возвращает не его, а пытается найти следующий по значению элемент. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
2. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в следующем порядке: сначала максимальный элемент, затем минимальный, затем второй максимальный, затем второй минимальный и т.д. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
3. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в следующем порядке: сначала минимальный элемент, затем максимальный, затем второй минимальный, затем второй максимальный и т.д. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
4. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в следующем порядке: сначала первый добавленный в массив элемент, затем последний, затем второй добавленный в массив элемент, затем предпоследний и т.д. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
5. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в следующем порядке: сначала последний добавленный в массив элемент, затем первый, затем предпоследний добавленный в массив элемент, затем второй и т.д. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
6. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в следующем порядке: сначала элемент, находящийся в середине массива, затем элемент справа от него, затем слева от него, затем элемент, находящийся на втором месте справа от среднего, затем – на втором месте слева от среднего и т.д. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
7. Описать шаблон класса для хранения данных (контейнер). Данные должны храниться, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый второй элемент, начиная с нулевого, то есть возвращал бы только элементы с чётными индексами. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
8. Описать шаблон класса для хранения данных (контейнер). Данные должны храниться, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый второй элемент, начиная с первого, то есть возвращал бы только элементы с нечётными индексами. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
9. Описать шаблон класса для хранения данных (контейнер). Данные должны храниться, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый третий элемент, начиная с нулевого, то есть возвращал бы только элементы с индексами, делящимися на 3. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
10. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый четвёртый элемент, начиная с нулевого, то есть возвращал бы только элементы с индексами, делящимися на 4. Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
11. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый второй элемент, начиная с нулевого, то есть возвращал бы только элементы с чётными индексами. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
12. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый второй элемент, начиная с первого, то есть возвращал бы только элементы с нечётными индексами. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.
13. Описать шаблон класса для хранения данных (контейнер). Данные должны хранится, например, в виде массива. Шаблон должен включать в себя функции добавления данных, удаления данных, вывода данных на консоль ( info() ). Реализовать итератор для перебора данных контейнера так, чтобы он перебирал элементы в порядке их добавления в контейнер. При этом перебирал бы каждый третий элемент, начиная с нулевого, то есть возвращал бы только элементы с индексами, делящимися на 3. При достижении последнего элемента итератор должен «переключаться» снова на первый элемент, то есть быть «кольцевым». Продемонстрировать работу шаблона на объектах как встроенного типа, так и собственного класса. Для этого создать собственный класс и перегрузить в нём нужные функции-операции.