

Машзал

1. Дополнить текст программы таким образом, чтобы все операторы функции main() работали корректно. Функцию main не менять, возможна только дополнительная печать. Тела перегруженных операторов не должны быть пустыми, должны содержать произвольные, но осмысленные действия.

```
class Arr { int * arr;
            int size;
public: Arr(int s, int date=0) : size(s){
            arr= new int[s];
            for(int i=0;i<size;arr[i++]=date);
        }
        ~Arr () { delete [] arr; }
};
int main() { Arr a1(15), a2(20,5), a3(30,3);  const Arr a4(5);
            cout<<a1 <<a2<<a3<<a4;
            a2 = a2 | 2; // объект a2 здесь не меняется
            a2 |=1; // a здесь меняется
            a3= (- a2) | (-a4);
            a3[5]=10;
            a1= a4 | a3;
            cout<<a1 <<a2<<a3<<a4;
            return 0;
        }
```

2. Включите в класс конструктор перемещения и оператор перемещения. Посмотрите, какие вызовы изменились.

ДЗ-2. Срок сдачи — до 11.03.23

1. АТД. Очередь для хранения данных.

На основе класса Список_элементов_произвольного_типа (использовать класс из ДЗ-1) определить класс **Очередь**, который **должен быть производным от Списка**. Элемент данных, помещаемых в очередь - тот же класс, что использовали в списке.

Максимальный допустимый размер очереди определен по умолчанию, а также может явно указываться при создании объекта-Очереди.

Определить необходимые конструкторы.
Обеспечить корректное уничтожение объектов.

В классе должны быть функции:
добавления элемента в конец очереди (**back**),
чтение первого элемента из очереди без его удаления (**front**) ,

удаление первого элемента очереди (**pop**),
проверка очереди на пустоту (**empty**),
текущее число элементов (**size**),
проверка, что очередь целиком заполнена (**full**).
вывод информации об элементах очереди без ее изменения (**print**).

Функции класса Список должны остаться прежними, а не переделываться под потребности очереди.

Через объект Очередь должны быть доступны только функции, указанные в задании. При этом базовый класс остается полноценным списком, т.е. в программе могут использоваться и объекты базового класса Список.

Глобальных переменных быть НЕ должно.

Использование STL и типа string запрещено.

Функция main должна демонстрировать работу с объектами написанных классов.

Пример работы: Date d1(...); Queue q1(5), q2; List l1; q1.back(d1); l1.push_back(d1); l1.print(); cout<<q2.size(); и т.д.