

Работа с DynArray. Часть 2.

Необходимо доработать написанный ранее класс DynArray <https://github.com/bugslayer312/TeachExamples/blob/master/templates/DynArray.h>, чтобы с ним можно было работать через итераторы как с другими контейнерами стандартной библиотеки. Для этого нужно:

1) добавить в него следующие методы:

```
template<class T>
DynArrayIterator<T> DynArray::begin();
```

```
template<class T>
DynArrayIterator<T> DynArray::end();
```

, которые возвращают начальный и конечный итераторы.

2) Добавить методы для вставки и удаления элементов из массива по заданной позиции итератора (insert/erase).

Для DynArrayIterator<T> можно определить синоним внутри класса:

```
typedef DynArrayIterator<T> iterator.
```

3) реализовать класс итератора

```
template<class T>
class DynArrayIterator;
```

Данный итератор будет позволять перебирать элементы массива. Для этого в нем должны быть реализованы методы:

T& operator*() – оператор разыменования указателя. Результат – значение текущего элемента массива, на который указывает итератор.

Операторы для перемещения на 1 позицию (префиксная и постфиксная формы):

```
DynArrayIterator<T> operator++()
```

```
DynArrayIterator<T> operator--()
```

Оператор для перемещения на произвольную позицию из текущей:

```
DynArrayIterator<T> operator+(int step)
```

```
DynArrayIterator<T> operator– (int step)
```

Операторы сравнения: == и !=.

4) Написать пример, демонстрирующий работу с контейнером.