

Линейные контейнеры. Динамический массив, стек, дек, очередь, циклический буфер

seagest

1 Динамически расширяющийся массив

Определение 1.1. Динамически расширяющийся массив — массивоподобный абстрактный тип данных, предлагающий следующий интерфейс:

- *push* n — добавить значение n в конец.
- *pop* — удалить значение из конца массива.
- *get* i — получить i -й элемент массива.

2 Списки

Определение 2.1. Список — последовательный набор узлов, предоставляющий следующий интерфейс:

Операция	Время	Примечание
Вставка в начало	$\mathcal{O}(1)$	
Удаление из начала	$\mathcal{O}(1)$	
Вставка в конец	$\mathcal{O}(1)$	Если двусвязный
Удаление из конца	$\mathcal{O}(1)$	Если двусвязный
Вставка в произвольное место	$\mathcal{O}(1)$	Если известно место
Удаление из произвольного места	$\mathcal{O}(1)$	Если известно место
Поиск	$\mathcal{O}(n)$	
Обращение по индексу	$\mathcal{O}(n)$	

2.1 Односвязный список

Определение 2.2. Односвязный список — последовательный набор из узлов с данными, где каждый узел знает, где лежит следующий за ним.

2.2 Двусвязный список

Определение 2.3. Двусвязный список — последовательный набор из узлов с данными, где каждый узел знает, где лежат следующий и предыдущий узлы.

3 Стек, дек и очередь

3.1 Стек

Определение 3.1. Стек (*stack*) — АД, который хранит элементы и предоставляет к ним доступ в рамках парадигмы LIFO (*Last in, First Out*).

3.2 Очередь

Определение 3.2. Очередь (*queue*) — АД, который хранит элементы и предоставляет к ним доступ в рамках парадигмы FIFO (*First in, First Out*).

3.3 Дек

Определение 3.3. Дек (*deque*) — АТД, который представляет из себя двустороннюю очередь, то есть можно вставлять/удалять в начало/конец.

4 Циклический буфер

Определение 4.1. Циклический буфер — это структура данных, использующая единственный буфер фиксированного размера таким образом, как будто бы после последнего элемента сразу же снова идет первый. С его помощью можно реализовать очередь на массиве (так структура выйдет *cache-friendly*).