

자료 구조 스터디 09

2021-2 KCA

배열의 다양한 표현과 성능

본 ppt의 자료는 Pt.J님의 자료와 김성열 교수님의 강의 및 여러 블로그를 참고하였음을 밝힙니다.

배열 (Array)

Index	→	0	1	2	3	4	5	6	7
Value	→	5	3	7	6	4	8	12	11

- 실제 메모리에는 Value 값들만 일렬로 저장됨
- Index는 값을 가리키기 위한 것.

배열을 저장하는 방법

- Packed vs. Unpacked
 - 배열을 앞에서부터 꽉 채워 나갈까?
 - 자리를 비워놓고 채울까?
- Sorted vs. Unsorted
 - item들이 삽입/삭제 시 정렬된 상태를 유지하는가?
- Item : 배열의 원소, 배열에 저장되는 data를 부르는 이름

Packed, Unsorted

Index	→	0	1	2	3	4	5	6	7
Value	→	5	3	7	6	4	X	X	X

- 간단하다. 들어오는 대로 앞에서부터 채워나가기.
- Item 개수 표시할 변수가 필요

Packed, Unsorted 성능 및 작동 방식

<성능>

Search : $O(n)$

Insert : [Search $O(n)$] 이후 $O(1)$

Delete : [Search $O(n)$] 이후 $O(1)$

보통 같은 값 중복 허용을 안하므로 Insert 할 때도 Search가 필요!

Delete 하고 빈 자리에 맨 마지막 원소를 넣으면 Packed 만족!

Packed, Sorted

Index	→	0	1	2	3	4	5	6	7
Value	→	3	5	7	8	10	X	X	X

- Sorted 되어 있으므로 Binary Search 가능!
- Search 성능은 높아지지만 Sorted 유지를 위해 밀고 당겨야 되어 Insert, Delete 성능이 안 좋아짐.

Packed, Sorted 성능 및 작동 방식

<성능>

Search : $O(\log n)$

Insert : [Search $O(\log n)$] 이후 $O(n)$

Delete : [Search $O(\log n)$] 이후 $O(n)$

보통 같은 값 중복 허용을 안하므로 Insert 할 때도 Search가 필요!

Insert, Delete 시에 Sorted를 유지하기 위해 당기거나 밀어야 한다!

I, D에 비해 Search가 월등히 많은 경우 자주 쓰인다. (학생부, 내비게이션, 사전 등)

Unpacked, Unsorted

Index	→	0	1	2	3	4	5	6	7
Value	→	5	3	9	6	4	8	11	1
Mark	→	O	O	X	O	O	X	O	X

- 빈 자리가 흩어져 있다.
- Item 별로 사용 중인지 표시 필요 => Mark

Unpacked, Unsorted 성능 및 작동 방식

<성능>

Search : $O(n)$

Insert : [Search $O(n)$] 이후 $O(1)$

Delete : [Search $O(n)$] 이후 $O(1)$

Delete : Data 건드리지 않고도 Mark만 X로 바꿔주면 됨.

Insert를 위해서는 비어있는 자리를 찾아야 한다.

Free List Head를 이용해 Insert를 빈 자리를 찾는 일이 $O(1)$ 에 가능해진다!

Free List Head (연결리스트의 응용)

Index	→	0	1	2	3	4	5	6	7
Value	→	5	3	-1	7	4	20	11	2
Mark	→	O	O	X	X	O	O	O	X

Free List Head → 3

- FLH 있는 index를 따라가면 마지막에 -1 들어있음.
- Insert 6 명령 -> index[3]의 value에 6 대입, FLH에 7 대입.
- Delete 11 명령 -> FLH를 6으로 덮어 쓰고, index[6]의 value를 3으로 수정.
- Linux 디스크 관리에서 같은 원리를 이용한 Free Block List 사용.

Unpacked, Sorted

Index	→	0	1	2	3	4	5	6	7
Value	→	3	5	7	8	11	12	15	17
Mark	→	O	O	O	X	X	O	X	O

- Mark가 X인 공간에도 값은 존재하므로 포함해서 Binary Search 가능! (Mark 가 X인 공간도 Sorted 유지되어야 함)
- Item 개수 표시하는 변수 필요하다.

Unpacked, Sorted 성능 및 작동 방식

<성능>

Search : $O(\log n)$

Insert : [Search $O(\log n)$] 이후 $O(n)$

Delete : [Search $O(\log n)$] 이후 $O(1)$

Insert 시 삽입 후 뒤의 값들을 하나씩 밀어줘야 한다. (빈 자리가 나올 때까지 밀면 되므로, Packed보다 빠르다)

Delete : Mark만 바꾼다.

DB에서 사용