

뉴·진·스

이분 탐색

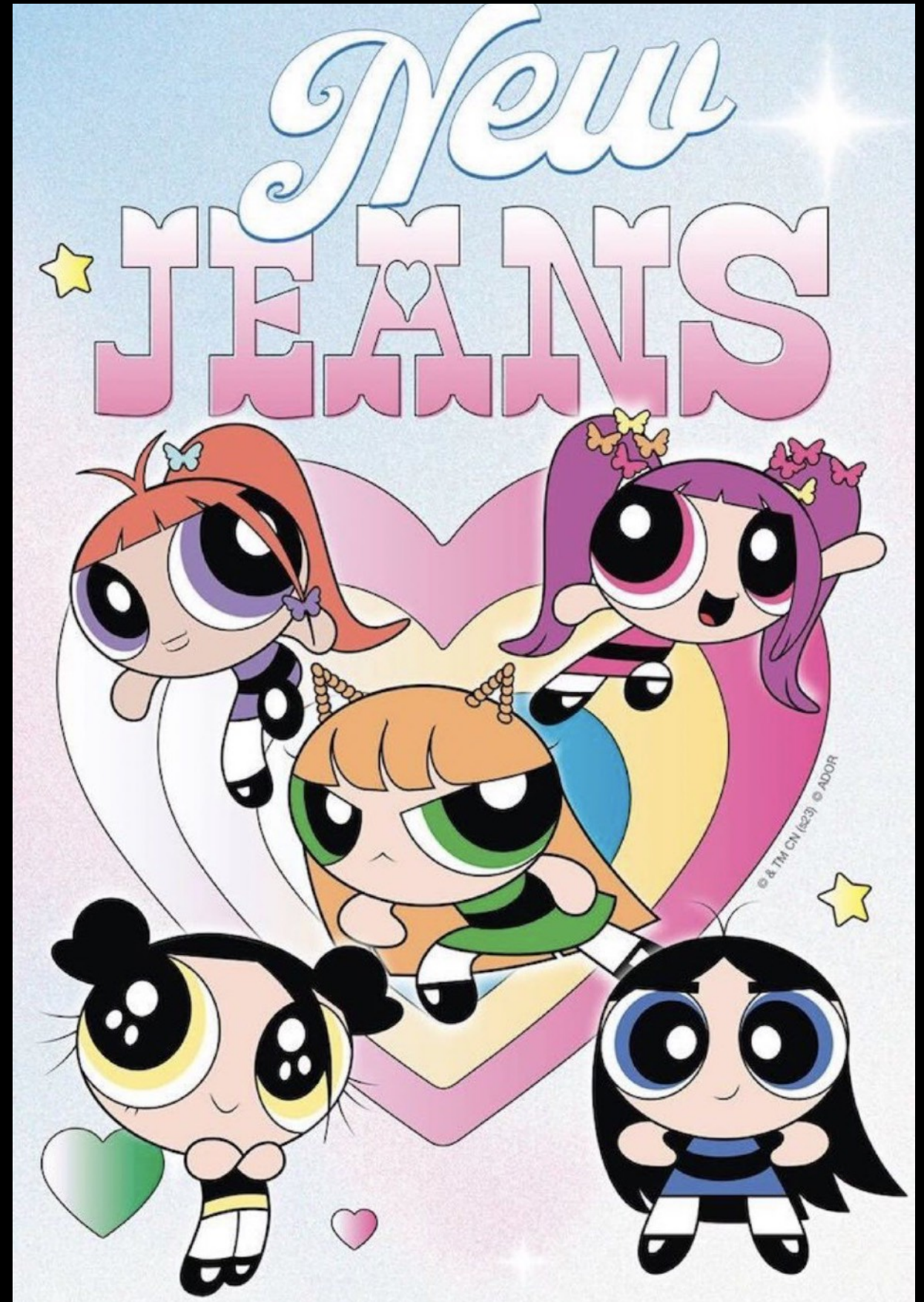
김준성 2023-11-01(화)

"사실 나도 잘 모름"

이번 스터디 중에서 전무후무한 야매 설명이 빈번히 나올 예정

목차

- 1.이분탐색
- 2.매개변수탐색
- 3.Live Coding



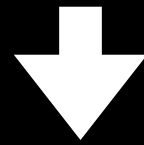
이분 탐색

이분 탐색이란?

- 원하는 값이 어디에 있는 지 찾아내는 알고리즘
- 정렬된 배열 또는 연속된 숫자의 상황에서 쓸 수 있습니다.
- 시간 복잡도는 $\log N$ 입니다.
- 이 알고리즘은 실생활에서 자주 사용되는 기법입니다.



숫자 2를 찾아보자!

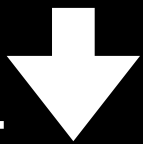


50

이분
탐색

0

100



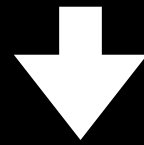
12

완전
탐색

0

100

숫자 62를 찾아보자!

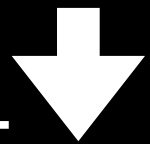


이분
탐색



0

100



완전
탐색



0

100

이분 탐색 구현

```
let sojuBottleCapNumber = Int.random(in: 1...100)
print("소주 병뚜껑 숫자: \(sojuBottleCapNumber)")

var count = 0
var left = 0
var right = 101
while (left + 1) < right {
    let mid = (left + right) / 2

    if mid <= sojuBottleCapNumber {
        left = mid
    } else {
        right = mid
    }
    count += 1
}
print("left: \(left), right: \(right), count: \(count)")
```

소주 병뚜껑 숫자: 21
left: 21, right: 22, count: 7

소주 병뚜껑 숫자: 76
left: 76, right: 77, count: 7

소주 병뚜껑 숫자: 71
left: 71, right: 72, count: 7

이분 탐색 구현

```
let sojuBottleCapNumber = Int.random(in: 1...100)
print("소주 병뚜껑 숫자: \(sojuBottleCapNumber)")

var count = 0
var left = 0
var right = 101
while left <= right {
    let mid = (left + right) / 2

    if mid < sojuBottleCapNumber {
        left = mid + 1
    } else {
        right = mid - 1
    }
    count += 1
}
print("left: \(left), right: \(right), count: \(count)")
```

소주 병뚜껑 숫자: 26
left: 26, right: 25, count: 7

소주 병뚜껑 숫자: 88
left: 88, right: 87, count: 7

소주 병뚜껑 숫자: 34
left: 34, right: 33, count: 7

이분 탐색 구현

```
let sojuBottleCapNumber = Int.random(in: 1...100)
print("소주 병뚜껑 숫자: \(sojuBottleCapNumber)")

var count = 0
var left = 0
var right = 101
while left <= right {
    let mid = (left + right) / 2

    if mid < sojuBottleCapNumber {
        left = mid + 1
    } else {
        right = mid - 1
    }
    count += 1
}
print("left: \(left), right: \(right), count: \(count)")
```

매개변수 탐색

매개변수 탐색이란?

- 이분 탐색에 변형을 준 알고리즘입니다.
- 이분 탐색의 조건문 부분을 함수화하면, 매개변수 탐색이 됩니다.
- lower bound와 upper bound는 각각 어떤 조건을 충족하는 범위의 최소값과 최대값을 알려줍니다.
- lower bound와 upper bound는 부등호가 ' \leq '이냐 ' $<$ '이냐에 따라 결정됩니다.

매개변수 탐색 구현

```
let sojuBottleCapNumber = Int.random(in: 1...100)
print("소주 병뚜껑 숫자: \(sojuBottleCapNumber)")
```

```
var count = 0
var left = 0
var right = 101
while left <= right {
    let mid = (left + right) / 2
```

```
func someFunction(parameter: SomeType) -> Bool
```

```
    if someFunction(mid) {
        left = mid + 1
    } else {
        right = mid - 1
    }
```

```
    count += 1
}
print("left: \(left), right: \(right), count: \(count)")
```

소주 병뚜껑 숫자: 26

left: 26, right: 25, count: 7

소주 병뚜껑 숫자: 88

left: 88, right: 87, count: 7

소주 병뚜껑 숫자: 34

left: 34, right: 33, count: 7

Lower-Upper Bound

```
// 10의 인덱스: 5, 6, 7
let array = [-10, 2, 3, 4, 8, 10, 10, 10, 777, 8888]
let ten = 10

var left = 0
var right = array.count - 1
while left + 1 < right {
    let mid = (left + right) / 2

    if ten <= array[mid] {
        right = mid
    } else {
        left = mid
    }
}

print("left: \(left), right: \(right)") // left: 4, right: 5
```

```
// 10의 인덱스: 5, 6, 7
let array = [-10, 2, 3, 4, 8, 10, 10, 10, 777, 8888]
let ten = 10

var left = 0
var right = array.count - 1
while left + 1 < right {
    let mid = (left + right) / 2

    if ten < array[mid] {
        right = mid
    } else {
        left = mid
    }
}

print("left: \(left), right: \(right)") // left: 7, right: 8
```

Lower-Upper Bound

```
// 10의 인덱스: 5, 6, 7
let array = [-10, 2, 3, 4, 8, 10, 10, 10, 777, 8888]
let ten = 10

var left = 0
var right = array.count - 1
while left <= right {
    let mid = (left + right) / 2

    if ten <= array[mid] {
        right = mid - 1
    } else {
        left = mid + 1
    }
}

print("left: \(left), right: \(right)") // left: 5, right: 4
```

```
// 10의 인덱스: 5, 6, 7
let array = [-10, 2, 3, 4, 8, 10, 10, 10, 777, 8888]
let ten = 10

var left = 0
var right = array.count - 1
while left <= right {
    let mid = (left + right) / 2

    if ten < array[mid] {
        right = mid - 1
    } else {
        left = mid + 1
    }
}

print("left: \(left), right: \(right)") // left: 8, right: 7
```


精神世界

쉬는 시간

1654 랜선 자르기



7795 먹을 것인가 먹힐 것인가



끝