## 알고리즘의 효율성 분석

이충기

명지대학교 컴퓨터공학과

## 문제 1: 입력 크기와 기본 연산

가장 맛을

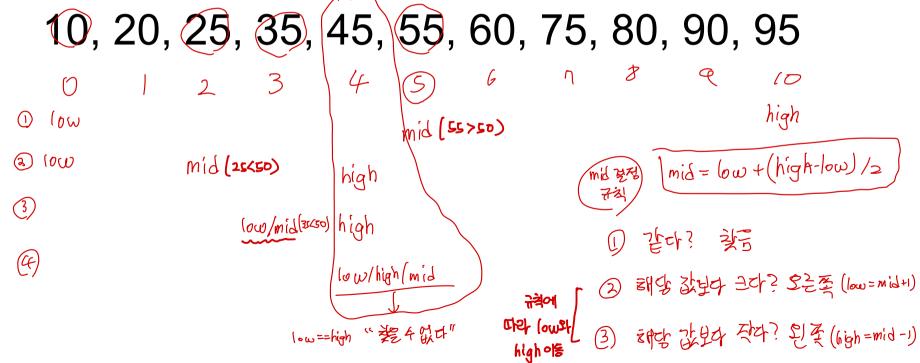
## 문제 2: 이진 탐색

다음의 숫자들에 대해 50를 이진탐색으로 찾는 과

정을 보여라.



총 11개 황수



- ① Hirth Geore, Rank 선언 ③ 製作 地方 Count 化过程 ス 3: 등수 마フフ
- 3 Score의 i(0~10+1)에 대해 Score[i] j(0~10+1)의 j를 즐거시켜 카더 \_ Score [i] 와 Score [ii)를 Alii
- 점수들을
- 3 Score Li] 가 어로 커서 Score Li) 와 Score Li) 가 같으면 Count 증가
  - 값이
- (# Score[i]의 비교가 끝았다면 (\*\*) i가 N서 까지 다 Count를 Ranh[i]에 저정하고 돈 특에는 좋죠 Count를 O으로 초기화,
  - 큰 순서대.

## <u>를 들면</u>

```
public static void main(String[] args){
    int count = 0;
    int Score[] = {82, 75, 98, 63, 40};
    int Rank[] = {1,2,3,4,5};

    for(int i = 0; i < 5; i++){
        if(Score[i] <= Score[j]){
            count++;
        }
    }
    Rank[i] = count;
    count = 0;
}

for(int i = 0; i < 5; i++){
    System.out.println(Rank[i] + " ");
}
</pre>
```

```
5명의 점수가 82
```

수는 2등, 3등, 1등

√(> 0)이다. 점수旨