

Euclidean

🕒 작성일시	@2022년 8월 4일 오후 5:39
▼ 강의 번호	
▼ 유형	
▼ 강사	
🔗 자료	
☑ 복습	<input type="checkbox"/>
📅 날짜	@2022년 8월 4일

유클리드 알고리즘 (Euclidean Algorithms)

- 최대공약수 구하는 알고리즘
- 약수들 중에서 가장 큰 수
- 반복구조를 이용하는 중요한 알고리즘

약수

3의 약수 - 1,3

4의 약수 - 1,2,4

5의 약수 - 1,5

6의 약수 - 1,2,3,6

8의 약수 - 1,2,4,8

12의 약수 - 1,2,3,4,6,12

공약수

공통된 약수

8과 12의 공약수 - 1,2,4

최대 공약수

두수의 공약수 중 최댓값

8과 12의 최대공약수 - 4

최대공약수 구하는 절차

먼저 어떤 복수의 수를 **소수의 곱셈** 형태로 분해하자. → 소인수 분해

$$8 = 1 \cdot 2 \cdot 2 \cdot 2$$

$$12 = 1 \cdot 2 \cdot 2 \cdot 3$$

그러나 이런 절차를 반복하는 것은 상당히 복잡하다.

어떤 수를 소인수 분해를 하려면 그 수 이하의 소수들을 모두 구해야 한다.

그리고 그 소수 중 작은 숫자부터 순서대로 원래의 수를 나누고, 나누어지지 않으면 그다음 소수의 순서로 계속 계산을 반복해야한다.

따라서 단순해 보이지만 절차는 상당히 복잡해진다.

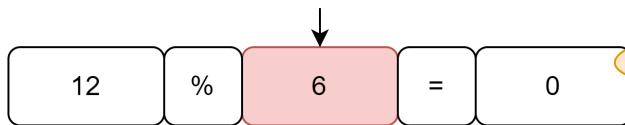
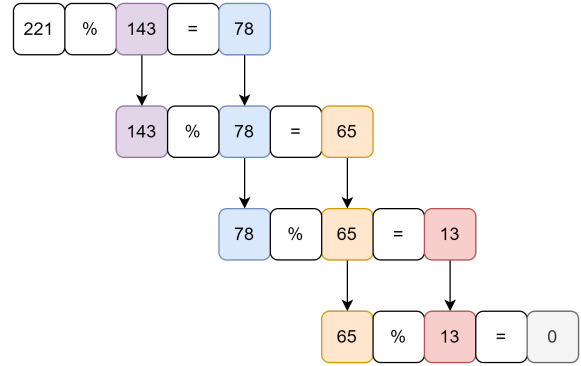
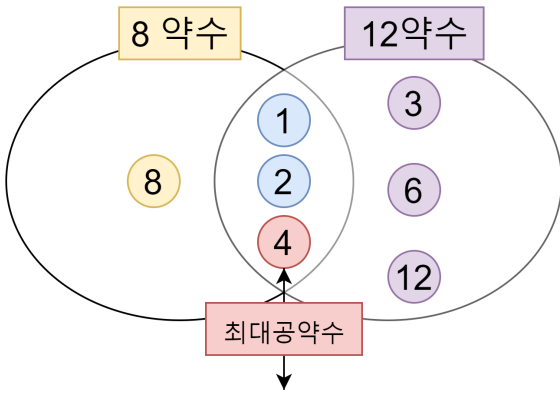
이러한 복잡성에 비해 매우 간단한 방법으로 최대공약수를 구하는 것이 바로 '**유클리드 알고리즘**'이다.

유클리드 알고리즘이란?

2300여년전 고대 그리스 수학자로 그가 생각해낸 수많은 수학적 이론중 하나가 유클리드 알고리즘 이다. 간단히 말하면 두 수의 나눗셈을 반복하여 최대공약수를 구하는 것 이다.

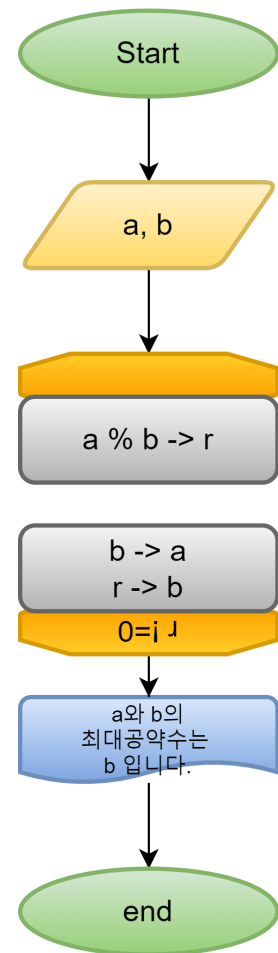
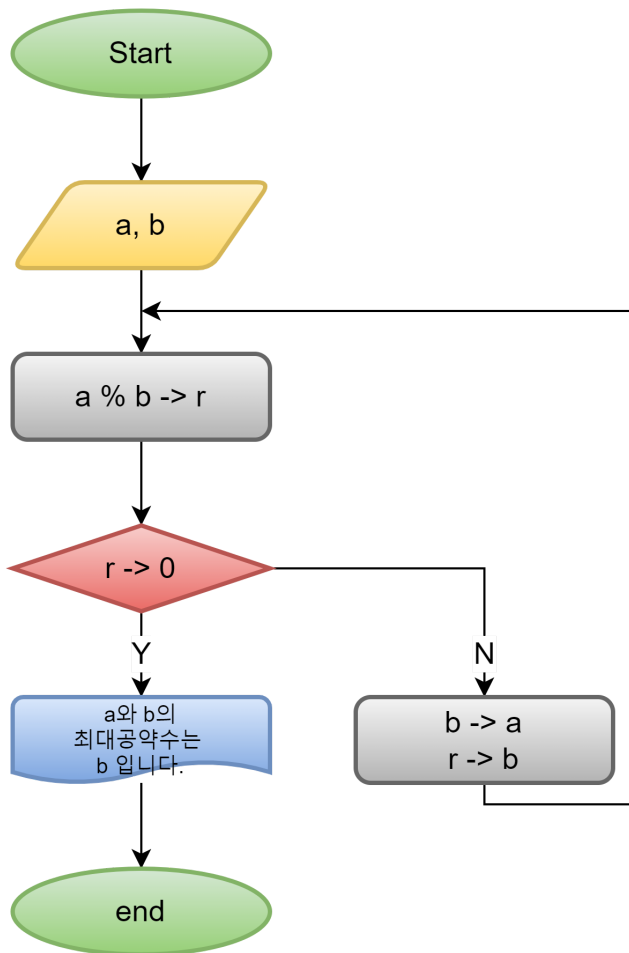
먼저 큰 수를 작은 수로 나눈다. 나머지가 나오지 않게 되면 (%) 그 때의 나누는데 사용된 작은 수가 최대 공약수 이다.

$12 \div 6 = 0$, 최대 공약수는 6이다.



나머지가 없이 딱 나누어 떨어지면
그 나누는데 사용된 수가
바로 최대 공약수 이다.

Flow chart



Java Code

```

package Eratos;
import java.util.*;
public class Eucild {

    public static void main(String[] args) {
        Scanner no = new Scanner(System.in);
        System.out.print("a를 입력하세요. >");
        int a = no.nextInt();
        System.out.print("b를 입력하세요. >");
        int b = no.nextInt();
        int r = 0;

        do{
            r = a % b;
            a = b;
            b = r;
        }while(r!=0);
    }
}
  
```

```
System.out.println("a 와 b 의 최대공약수는" + a +"입니다." );  
}  
}
```