



# **자료구조 (Data Structure)**

---

## **Programming Assignment 2**

서강대학교  
컴퓨터공학과/인공지능학과

- **문제1:** 집합  $S$ 는  $\{a, b, c, \dots\}$ 와 같이 소문자 알파벳 순으로 이루어졌다고 하자. (예: If  $|S|=3$ ,  $S=\{a,b,c\}$ . If  $|S|=5$ ,  $S=\{a,b,c,d,e\}$ .) 집합  $S$ 의 부분집합들은 각  $S$ 의 원소에 대하여 포함하느냐 그렇지 않느냐로 구분된다. 0을 포함하지 않을 경우 1을 포함하는 경우라 할 때에 가능한  $S$ 의 부분집합은 다음과 같다: 0 1 0 1 0 1 0 1, 0 0 1 1 0 0 1 1, 0 0 0 0 1 1 1 1, etc. 즉, 위의 0과 1을 가지고  $S$ 의 원소의 개수만큼의 자리수를 가진 2진수로 표현한다면 모든  $S$ 의 powerset  $P(S)$ 를 표현할 수 있다. 이를 이용하여, 사용자에게  $S$ 의 원소의 수를 입력받아,  $P(S)$ 를 모두 출력하는 recursive 함수 `PowerSet( )`을 작성하시오.
- 입력, 출력 형식 자유, 함수의 인자 자유.
- 예) 입력 3일 때,  $\{\}$   $\{a\}$   $\{b\}$   $\{c\}$   $\{a\ b\}$   $\{a\ c\}$   $\{b\ c\}$   $\{a\ b\ c\}$ 과 같이 **순서에 맞게** 출력 필요 (즉 공집합, 원소의 수 1인 집합들, 2인 집합들, ....., 원소의 수  $n$ 인 집합.  $n > 0$  가정.



# HW2

## ■ 문제 2:

강의 자료의 `pmatch()` 함수는 KMP algorithm을 사용하여 주어진 string내에 pattern이 존재하는지를 찾아주는 함수이다. 만약 string에서 pattern이 발견되면, `pmatch()`는 가장 처음 pattern과 string이 matching되는 부분의 starting index를 return한다.

`pmatch()`를 수정하여, string과 pattern이 matching되는 모든 부분의 starting indices를 출력하는 함수 `pmatch_all()`을 작성하라.



# HW2

## ■ 문제 2:

예제	
입력	출력
bbbbbabbbbbc bbb	0 1 2 6 7 8
bbbbbabbbbbc aa	

# HW2

- 문제 2:

- 입력:

- 첫 번째 줄-string
    - 두 번째 줄-pattern
    - 각 입력의 길이는 최소 1글자, 최대 30글자 (그 외의 경우는 들어오지 않음)

- 출력:

- matching되는 모든 부분의 starting index를 한 줄에 하나씩 출력
    - matching되는 부분이 없을 경우 아무 것도 출력하지 않음

# HW2

- 문제 2:
  - 제약 조건:
    - 모든 입출력은 stdin, stdout 사용
    - 전역변수 사용 불가
    - 정적할당 가능

# HW2

## ■ 문제 3:

Array을 parameter로 받은 후, 연속된 숫자만이 저장되어 있는지 확인하는 함수 `check_array()`를 만들고자 한다. 예를 들어, {4, 1, 5, 2, 3}와 같은 array의 경우에는 1부터 5까지의 연속적인 숫자들이 저장되어 있으므로, `check_array()`가 1을 return하도록 한다. 그러나 {10, 14, 12, 15, 11, 9}와 같은 array의 경우에는 중간에 13이 빠져 있으므로 `check_array()`가 0을 return하도록 한다.

`check_array()`를 구현하는 데에는 여러 가지가 있지만, 가장 쉬운 방법은 배열을 먼저 sorting한 후 중간에 빠진 숫자가 있는지 찾는 것이다. 이 경우, array의 크기를  $n$ 이라 할 때, 함수의 시간 복잡도는  $O(n \log n)$ 이 된다.

물론 이보다 더 나은 방법도 존재한다.  **$O(n)$ 의 시간 복잡도를 가지도록** `check_array()`를 작성해보자.



# HW2

## ■ 문제 3:

예제	
입력	출력
5 4 1 5 2 3	1
6 10 14 12 15 11 9	0
1 1	1
4 2 2 3 1	0





# HW2

- 문제 3:

- 입력:

- 첫 번째 줄-원소의 개수  $n$   
두 번째 줄- $n$ 개의 원소
    - $n$ 과 각각의 원소의 크기는 1 이상 100 이하

- 출력:

- 배열이 연속된 숫자로만 이루어져 있을 경우 1, 아니면 0



# HW2

- 문제 3:

- 제약 조건:

- 모든 입출력은 stdin, stdout 사용
    - 전역변수 사용 불가
    - 정적할당 사용 불가
    - 완성된 함수의 시간 복잡도는  $O(n)$ 이어야 함.



# HW2

- 문제 4:

자료구조 수강생 명단을 array를 사용하여 관리하고자 한다. student.txt에서 수강생 명단을 읽어온 후, array에 저장하고, lexical order에 따라 sorting하는 프로그램을 작성하라.



# HW2

## ■ 문제 4:

예제	
입력 (student.txt)	출력
6 Kim Minsu Kim Minju Choi Hojeong Cho Yujin Lee Minsu Choi Minjeong	Cho Yujin Choi Hojeong Choi Minjeong Kim Minju Kim Minsu Lee Minsu



# HW2

- 문제 4:
  - 입력 (student.txt):
    - 첫 줄에 학생의 수
    - 다음 줄부터 한 줄에 하나씩 학생들의 이름
  - 출력:
    - Sorting된 학생들의 이름 (한 줄에 하나씩)



# HW2

- 문제 4:

- 제약 조건:


- File input, stdout 사용
    - 전역변수 사용 불가
    - 정적할당 사용 불가
    - string.h 사용 불가
    - sorting 함수 직접 구현할 것
    - Lexical order에 따라 정렬 (성으로 비교한 후, 성이 같을 경우 이름으로 비교)



# 제출 주의사항

- 제출 화일에 대한 설명은 사캠 공지 참조.
- 컴파일 에러가 발생할 경우 0점 처리
- 무한 루프/세그멘테이션 오류는 해당 testcase 0점 처리
  - 프로그램이 일정시간 안에 답을 출력 안하는 경우 틀린 출력이라고 간주함.
- 입출력 양식이 틀릴 경우 감점



- 
- 보고서 (한글로 작성하되, 모든 컴퓨터 언어 및 자료구조 관련 명사는 영어로 작성)
    - 이름: Document.pdf
    - 반드시 pdf 파일로 변환 후 제출 (pdf 이 아닌경우 0점 처리)