

## 2021년 1학기 자료구조론 과제 #2

### ■ 과제내용 및 명세

#### ▲ 문제 : Binary Search Tree 구현

- 교재 Binary Search Trees 참고
- Insert, Delete, Search 기능 구현
- Data type : positive integer
- 파일 입·출력
- 사용 언어 : C
- ※ Visual Studio 권장

#### ▲ 명세(Specification)

- 입출력 : 파일로 Binary Search Tree의 elements를 입·출력 한다.
- 입출력용 파일 명

입력	출력
input.txt	output.txt

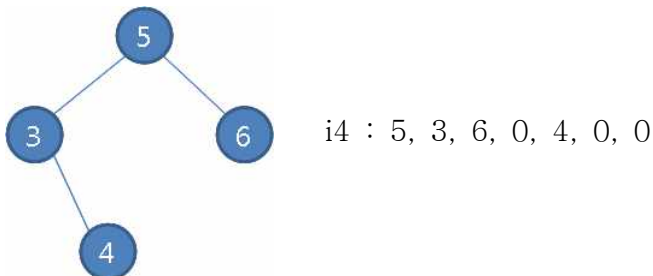
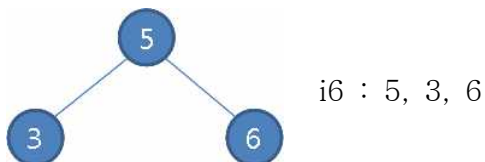
- \* 간혹 C:\W나 특정 폴더 내에 입력 혹은 출력 파일을 생성하도록 하는 분이 계시는데, 상대 폴더 위치로 지정하셔서 자신의 폴더 내에 해당 파일이 생성되도록 해 주시기 바랍니다. (단순히 “input.txt”나 “output.txt”로 지정하면 소스 프로그램과 같은 폴더 내에 생성됩니다.)
- 파일 명은 반드시 위에 주어진 이름을 사용하고 불가피할 경우 사유와 사용한 각 파일에 대한 설명을 문서에 포함하도록 한다.
- 입출력 파일의 형식은 아래의 예와 동일한 형태를 가져야한다. 아래의 예와 다르게 할 경우 감점사항에 해당하며 입출력 파일의 형식에 대한 상세한 설명을 문서에 포함해야 한다.
- 입력 형식 : □ 는 data를 의미
  1. insertion : i□ (중복 시 left child에 삽입)
  2. deletion : d□ (중복 시 가장 먼저 탐색 되는 것 삭제)
  3. search : s□ (**preorder** 탐색)
- 입력 파일 예 (input.txt)

파일 형식	입력의 최종 결과 (참고)
i5 i3 i6 i4 i7 s5 i9 d6 i1	<pre>graph TD; 5((5)) --- 3((3)); 5 --- 7((7)); 3 --- 1((1)); 3 --- 4((4)); 7 --- 9((9))</pre>

※ input 방법을 반드시 document나 source code내에 명시 할 것.

- 출력 형식("해당 입력 command : tree")

1. 입력 파일의 명령과 결과 tree를 **level-order**로 출력하도록 한다.
2. level order로 순회 시 비어있는 노드의 경우 0을 출력 (아래 예시 참고)
3. Search의 경우 존재하면 "Exist", 존재하지 않으면 "Not Exist" 출력
4. Delete의 경우 Tree에 존재하지 않는 data면 "Not Exist" 출력



- 출력 파일 예 (output.txt)

입력 파일의 각 command에 대한 각각의 결과를 단계마다 모두 출력한다.

파일 형식
i5 : 5
i3 : 5, 3, 0
i6 : 5, 3, 6
i4 : 5, 3, 6, 0, 4, 0, 0
i7 : 5, 3, 6, 0, 4, 0, 7
s5 : Exist
i9 : 5, 3, 6, 0, 4, 0, 7, 0, 0, 0, 0, 0, 0, 9
d6 : 5 3 7 0 4 0 9
d2 : Not Exist

◦ 기본구현 :

- Binary Search Tree 구현
- Insert, Delete, Search 기능 구현

### ▲ 채점기준

- 각 기능의 정확한 동작
- Documentation

## ■ 과제제출

### ▲ 제출기한 : 2021년 5월 23일 (일)

※ 2021년 5월 23일(일) 자정까지 upload된 과제는 인정합니다.

### ▲ 제출해야 하는 결과물 : source, document, data

1. source(.c, .cpp 파일) : 상세한 comment를 달 것
2. document (파일이름 : "과제3\_학번\_이름.doc or .hwp")
  - △ 과제 내용 간략하게 요약
  - △ 구현환경 : 구현에 사용한 언어와 compiler, compile 방법 등을 포함
  - △ 설명 : source의 구조와 입·출력 방법 및 Binary Search Tree 자료구조와 각각의 function에 대한 상세한 설명
  - △ 자신이 실행해 본 sample 입력과 결과를 document에 첨부
  - △ 기타 : 참조한 source나 구현하지 못한 기능이 있을 경우 언급할 것
3. input : 자신이 구현한 모든 함수들을 테스트 해 볼 수 있는 입력 파일 ("input.txt")
  - 예시의 입력파일과 동일한 형식
4. output : 입력파일에 대한 예상 결과물 ("exp\_output.txt")
  - 프로그램을 실행해서 나온 결과물("output.txt")
  - 예시의 출력파일과 동일한 형식

### ▲ 제출방법 :

1. 자신의 학번과 이름으로 새 폴더 생성 (예:2020004005 홍길동)
2. source code와 document, data(input, output)를 폴더에 넣은 후, 폴더 자체를 압축 (virus check 권장)
3. 블랙보드에 '[과제3]2020004005\_홍길동'으로 과제번호와 자신의 이름을 명시한 후 제출. 잘못 upload하여 다시 올릴 경우나 더 추가하여 올릴 경우에는 '[과제3] 2020004005\_홍길동\_ver2'식으로 version관리를 해 주시기 바랍니다. (최근 version으로만 채점하겠습니다.)

▲ 위의 제출 방법을 지키지 않은 경우 (제출물을 압축하지 않은 경우, 제출 파일의 이름이 다른 경우 등) 감점 사항에 해당합니다.

## ■ 주의사항

▲ 동일한 코드임이 확인될 경우, 같은 코드를 제출한 학생들 모두 0점 처리되니 주의 바랍니다.

▲ 「명세」와 「과제제출」에 명시된 것을 반드시 준수하시기 바랍니다. (지켜지지 않았을 경우 감점)

▲ 알고리즘들의 구현 중 참조한 source가 있을 경우 document에 그 출처와 참조 정도를 반드시 언급해 주시기 바랍니다.

▲ 추가 질문사항은 하단에 명시된 담당조교에게 email로 제출해 주시기 바랍니다.

담당조교 : 공업센터본관 303호 임베디드보안시스템 연구실 정혜민(hmjung2@hanyang.ac.kr)