

## 프로그래밍 과제 #2 : Partition Problem

2018. 6. 5.

**문제]** 강의자료 7장 26페이지에 제시된 Partition 문제를 해결하는 C Program을 Backtracking 알고리즘을 이용하여 구현하시오. (반드시 C로 작성할 것. C++ 사용 불가)

The Partition (to equal sized subset) Problem (분할)

Input : A set S of integers.

Question : Is it possible to partition S into two subsets  $S_1$  and  $S_2$  such that the sum of the numbers in  $S_1$  equals the sum of the numbers in  $S_2$ ?

Example

$S = \{20, 30, 40, 80, 90\}$ .

The answer is "yes" since there are equal sized subsets

$S_1 = \{20, 30, 80\}$  and  $S_2 = \{40, 90\}$ .

**절차]** C 프로그램의 구동 절차는 다음과 같다.

(1) 파일 input.txt를 통해 테스트 케이스를 입력 받는다. 파일의 맨 첫줄에는 총 테스트 케이스의 개수가 주어진다. 이후 각 테스트 케이스에 해당하는 집합의 원소 개수와 원소가 두 줄에 걸쳐 차례로 주어진다.

(2) 각 테스트 케이스에 대하여 partition이 가능하면 'Partition is possible'과 그 해집합을 화면에 출력(원소 출력 순서는 무관)하고, 분할 가능하지 않으면 'Partition is not possible'을 화면에 출력한다.

예1)  $S = \{3, 1, 5, 9, 12\}$

Partition is possible  
1, 5, 9 / 3, 12

$S = \{3, 1, 5, 8, 12\}$

Partition is not possible

(3) 각 테스트 케이스를 해결하기 위해 배열을 선언할 시에는 반드시 동적 할당을 사용한다. 입력은 반드시 input.txt로부터 받고 출력은 화면(stdout)에 하여야 한다. backtracking 구현에는 recursive 함수를 사용한다. 이 모든 사항을 지키지 않은 코드는 0점 처리한다. 정상적으로 작동하지 않는 코드 역시 0점 처리 한다.

(4) 채점은 Linux gcc compiler를 이용하여 이루어진다. 제출한 코드가 특정한 compiler 환경에서만 구동되는 경우(필수 라이브러리를 include 하지 않았음에도 정상작동 하는 경우, compile error나 segmentation error가 나야 함에도 compiler가 이를 적절히 처리하는 경우 등)에는 0점 처리한다. Linux 계정이 없어 이를 확인하지 못하는 경우에는 (5)를 참조한다.

(5) Linux 환경을 사용하지 못하는 경우에는 Dev C++ 을 이용하여 프로그램을 작성하거나, 타 환경에서 작성한 코드를 [https://www.onlinegdb.com/online\\_c\\_compiler](https://www.onlinegdb.com/online_c_compiler) 을 이용하여 테스트할 수 있다. 해당 online compiler 사용 시 텍스트 파일 첨부은 좌측 상단 new file 버튼을 이용하여 할 수 있다. 자세한 사용법은 수업시간에 별도로 안내한다.

\*제출 관련 사항

-제출일 : 2018. 6. 24 (일) 밤 11시 59분까지

-제출 방법 :

파일명 : snnnnnn\_HW2.c (여기서 nnnnnn여섯자리는 학번)

a) 압축파일 제목 : snnnnnn\_HW2.zip

b) 제출 파일을 압축하여 사이버 캠퍼스 과제란에 업로드하여 제출

c) 테스트케이스 파일은 별도로 제출할 필요 없으며, 제출 방법을 지키지 않을 시 10% 감점 한다.