

# **Sum of Subsets Problem**

## **With Backtracking algorithm**

소프트웨어학부  
20170294 박해영

## < 목 차 >

1. 목표	-----	3p
2. Problem & Input/Output	-----	3p
3. 구현 Language & 사용 Tool	-----	3p
4. 교재의 입력 데이터 테스트	-----	4p
5. 자작 입력 데이터 생성 & 알고리즘 과정 손계산	-----	5p
6. 자작 입력 데이터 테스트	-----	7p

## 1. 목표

되추적 알고리즘(backtracking algorithm)은 상태공간 트리와 유망 함수를 통해 풀고자 하는 형태이다. 되추적 알고리즘은 마디가 유망하고 그 마디에 해답이 없는 경우에만 그 마디의 자식마디를 방문하는 것을 제외하고는 깊이 우선 검색과 같다. 다음 마디가 유망함을 판단하는 조건을 알아내고, 이를 상태공간 트리를 이용하여 풀어냄으로써 되추적 알고리즘의 기본 문제 해결 방법을 알고자 한다.

## 2. Problem & Input / Output

- \* **Problem** :  $n$ 개 양의 정수(무게)와 양의 정수  $W$ 를 가지고, 합이  $W$ 가 되는 모든 정수의 조합
- \* **Input** : 양의 정수  $n$ , 감소하지 않는 순으로 정렬된 양의 정수의 배열  $w$ , 양의 정수  $W$
- \* **Output** : 합이  $W$ 가 되는 모든 정수의 조합

## 3. 구현 Language & 사용 Tool

- \* **구현 언어** : C ++ language
- \* **사용 Tool** : Visual Studio 2015

## 4. 교재의 입력 데이터 테스트

### 1) Ex5.9

```
Input number of Item : 4
Input number of maximum weight : 13
Input weight of Item w1 : 3
Input weight of Item w2 : 4
Input weight of Item w3 : 5
Input weight of Item w4 : 6

< Sum of subsets >

include w1 : yes
include w2 : yes
include w3 : no
include w4 : yes
```

## 5. 자작 입력 데이터 생성 & 알고리즘의 과정 손계산

Weight 개수  $n = 4$ ,  $W = 18$

$W$ 

2	5	6	7
---	---	---	---

$i = 0$ , weight = 0, total = 20

Promising (0)

$0 + 20 \geq 18$  &&  $0 + 2 \leq 18$  ; true

include[1] = Yes

Sum-of-subsets (1, 0+2, 20-2) ;

$i = 1$ , weight = 2, total = 18

Promising (1)

$2 + 18 \geq 18$  &&  $2 + 5 \leq 18$  ; true

include[2] = Yes

Sum-of-subsets (2, 2+5, 18-5) ;

$i = 2$ , weight = 7, total = 13

Promising (2)

$7 + 13 \geq 18$  &&  $7 + 6 \leq 18$  ; true

include[3] = Yes

Sum-of-subsets (3, 7+6, 13-6) ;

$i = 3$ , weight = 13, total = 7

Promising (3)

$13 + 7 \geq 18$  &&  $13 + 7 \leq 18$  ; false

return

include[3] = No

Sum-of-subsets (3, 7, 13-6) ;

$i = 3$ , weight = 7, total = 7

Promising (3)

$7 + 7 \geq 18$  ; false return

include[2] = No

Sum-of-subsets (2, 2, 18-5) ;

$i = 2$ , weight = 2, total = 13

Promising (2)

$2 + 13 \geq 18$  ; false return

include[1] = No

Sum-of-subsets (1, 0, 20-2) ;

$i = 1$ , weight = 0, total = 18

Promising (1)

$0 + 18 \geq 18$  &&  $0 + 5 \leq 18$  ; true

include[2] = Yes

Sum-of-subsets (2, 0+5, 18-5) ;

$i = 2$ , weight = 5, total = 13

Promising (2)

$5 + 13 \geq 18$  &&  $5 + 6 \leq 18$  ; true

include[3] = Yes

Sum-of-subsets (3, 5+6, 13-6) ;

$i = 3$ , weight = 11, total = 7

Promising (3)

$$11+7 \geq 18 \quad \&\& \quad 11+7 \leq 18$$

include[4] = Yes

Sum-of-subsets (4, 11+7, 7-7)

i=4, weight=18, total=0

Promising (4)

$$18+0 \geq 18 \quad \&\& \quad 18 == 18 \Rightarrow \text{true}$$

weight == w : true

출력: Include w1 : No

Include w2 : Yes

Include w3 : Yes

Include w4 : Yes

include[4] = No

Sum-of-subsets (4, 11, 7-7)

i=4, weight=11, total=0

Promising (4)

$$11+0 \geq 18 \Rightarrow \text{false} \Rightarrow \text{return}$$

include[3] = No

Sum-of-subsets (3, 5, 13-6)

i=3, weight=5, total=7

Promising (3)

$$5+7 \geq 18 \Rightarrow \text{false} \Rightarrow \text{return}$$

include[2] = No

Sum-of-subsets (2, 0, 18-5)

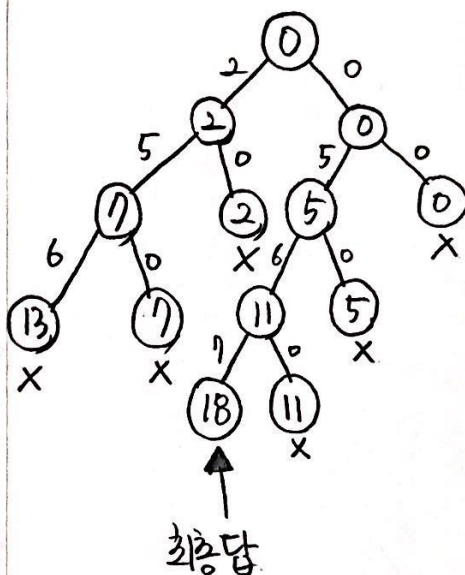
i=2, weight=0, total=13

Promising (2)

$$0+13 \geq 18 \Rightarrow \text{false} \Rightarrow \text{return}$$

⇒ 항등조건

상태공간 트리



정의수 1개

출력값:

Include w1 : No ... 2

Include w2 : Yes ... 5

Include w3 : Yes ... 6

Include w4 : Yes ... 7

18

## 6. 자작 입력 데이터 테스트

```
Input number of Item : 4  
Input number of maximum weight : 18  
Input weight of Item w1 : 2  
Input weight of Item w2 : 5  
Input weight of Item w3 : 6  
Input weight of Item w4 : 7
```

< Sum of subsets >

```
include w1 : no  
include w2 : yes  
include w3 : yes  
include w4 : yes
```

