

# 2019년 알고리즘

- HW 10 -

제출일자	2019.11.13.
이름	장수훈
학번	201402414
분반	01

## 1 Knapsack problem

```
public static void main(String[] args) {  
  
    FileReader fr;  
    try {  
        fr = new FileReader("C:\\Users\\micke\\eclipse-workspace\\ALG\\src\\data10_knapsack.txt");  
        BufferedReader br = new BufferedReader(fr);  
        String line = br.readLine();  
        int[] input_item = new int[100];  
        int[] input_value = new int[100];  
        int[] input_weight = new int[100];  
  
        int index = 0;  
  
        while (line != null) {  
            StringTokenizer st = new StringTokenizer(line, ",");  
            while (st.hasMoreTokens()) {  
                input_item[index] = Integer.parseInt(st.nextToken());  
                input_value[index] = Integer.parseInt(st.nextToken());  
                input_weight[index] = Integer.parseInt(st.nextToken());  
                index++;  
            }  
            line = br.readLine();  
        }  
        int[] itemNum = new int[index];  
        int[] value = new int[index];  
        int[] weight = new int[index];  
  
        for (int i = 0; i < index; i++) {  
            itemNum[i] = input_item[i];  
            value[i] = input_value[i];  
            weight[i] = input_weight[i];  
        }  
  
        Scanner input = new Scanner(System.in);  
        System.out.print("배낭의 사이즈를 입력하세요(0~50) :");  
        int size = input.nextInt();  
  
        if (size < 0 || size > 50) {  
            System.out.println("사이즈 설정이 잘못되었습니다.");  
        } else {  
            int[][] knap = OPT(value, weight, size);  
            print(knap, weight, value, size);  
        }  
    } catch (IOException e) {  
    }  
}
```

입력 받는 부분이다. item, value, weight를 각각 배열에 넣어주는 부분이다. 배낭의 사이즈는 0~50 사이로 받도록 하였다.

## OPT 와 출력하는 함수 각각 구현

$$OPT(i, w) = \begin{cases} 0 & \text{if } i = 0 \\ OPT(i-1, w) & \text{if } w_i > w \\ \max\{OPT(i-1, w), v_i + OPT(i-1, w - w_i)\} & \text{otherwise} \end{cases}$$

OPT함수를 구현 ↓

```
public static int[][] OPT(int value[], int weight[], int size) {
    int[][] table = new int[weight.length + 1][size + 1];

    for (int j = 0; j <= size; j++) {
        for (int i = 0; i <= weight.length; i++) {
            table[i][j] = 0;
        }
    }

    for (int i = 1; i <= weight.length; i++) {
        for (int w = 1; w <= size; w++) {
            if (weight[i - 1] > w) {
                table[i][w] = table[i - 1][w];
            } else {
                table[i][w] = max(table[i - 1][w], value[i - 1] + table[i - 1][w - weight[i - 1]]);
            }
        }
    }

    return table;
}
```

출력하는 함수 구현 ↓

```
public static void print(int[][] knap, int[] weight, int[] value, int size) {
    int n = weight.length;
    int w = size;
    boolean[] isitem = new boolean[weight.length + 1];

    for (int i = 0; i < weight.length + 1; i++) {
        for (int j = 0; j < size + 1; j++) {
            System.out.printf("%3d", knap[i][j]);
        }
        System.out.println();
    }

    System.out.println("max : " + knap[weight.length][size]);

    while (n > 0 && w > 0) {
        if (knap[n][w] != knap[n - 1][w]) {
            isitem[n] = true;
            w = w - weight[n - 1];
        }
        n--;
    }

    System.out.print("item : ");
    for (int i = 1; i < weight.length + 1; i++) {
        if (isitem[i] == true)
            System.out.print(i + " ");
    }
}
```

## 결과

```
<terminated> Knapsack [Java Application] C:\Program Files\Java\jre1.8.0_221\bin\javaw.exe (2019. 11. 13. 오후 5:08:36)
배낭의 사이즈를 입력하세요 (0~50) : 11
0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 1 1 1 1
0 1 6 7 7 7 7 7 7 7 7 7
0 1 6 7 7 18 19 24 25 25 25 25
0 1 6 7 7 18 22 24 28 29 29 40
0 1 6 7 7 18 22 28 29 34 35 40
max : 40
item : 3 4
```

## 2 Sequence Alignment

x : tcatagttaaca / y : tcagaagtacc

서열	X		t	c	a	t	a	g	t	t	a	a	c	a
y	Inde	0	1	2	3	4	5	6	7	8	9	10	11	12
	0	-10	-2	-4	-6	-8	-10	-12	-14	-16	-18	-20	-22	-24
		-2	-4	-6	-8	-10	-12	-14	-16	-18	-20	-22	-24	-26
t	1	-2	-4	-6	-8	-10	-12	-14	-16	-18	-20	-22	-24	-26
		-4	-6	-8	-10	-12	-14	-16	-18	-20	-22	-24	-26	-28
c	2	-4	-6	-8	-10	-12	-14	-16	-18	-20	-22	-24	-26	-28
		-6	-8	-10	-12	-14	-16	-18	-20	-22	-24	-26	-28	-30
a	3	-6	-8	-10	-12	-14	-16	-18	-20	-22	-24	-26	-28	-30
		-8	-10	-12	-14	-16	-18	-20	-22	-24	-26	-28	-30	-32
g	4	-8	-10	-12	-14	-16	-18	-20	-22	-24	-26	-28	-30	-32
		-10	-12	-14	-16	-18	-20	-22	-24	-26	-28	-30	-32	-34
a	5	-10	-12	-14	-16	-18	-20	-22	-24	-26	-28	-30	-32	-34
		-12	-14	-16	-18	-20	-22	-24	-26	-28	-30	-32	-34	-36
a	6	-12	-14	-16	-18	-20	-22	-24	-26	-28	-30	-32	-34	-36
		-14	-16	-18	-20	-22	-24	-26	-28	-30	-32	-34	-36	-38
g	7	-14	-16	-18	-20	-22	-24	-26	-28	-30	-32	-34	-36	-38
		-16	-18	-20	-22	-24	-26	-28	-30	-32	-34	-36	-38	-40
t	8	-16	-18	-20	-22	-24	-26	-28	-30	-32	-34	-36	-38	-40
		-18	-20	-22	-24	-26	-28	-30	-32	-34	-36	-38	-40	-42
a	9	-18	-20	-22	-24	-26	-28	-30	-32	-34	-36	-38	-40	-42
		-20	-22	-24	-26	-28	-30	-32	-34	-36	-38	-40	-42	-44
c	10	-20	-22	-24	-26	-28	-30	-32	-34	-36	-38	-40	-42	-44
		-22	-24	-26	-28	-30	-32	-34	-36	-38	-40	-42	-44	-46
c	11	-22	-24	-26	-28	-30	-32	-34	-36	-38	-40	-42	-44	-46
경로 1		t	c	a	t	a	g	t	t	a	a	c	a	점수 : 1
(위쪽)		t	c	a	g	a	a	g	t	-	a	c	c	
경로 2		t	c	a	t	a	g	t	t	a	a	c	a	점수 : 1
(가운데)		t	c	a	g	a	a	g	t	a	-	c	c	
경로 3		t	c	a	t	a	g	t	t	a	a	c	a	점수 : 1
(아래쪽)		t	c	a	g	a	a	g	t	a	c	c	-	