

<알고리즘 Handwriting Assignment #3> 202010819 조정현

1.

```
a) 29 ∵ 1 + 4 × 3 + 2 + 3 × 4 + 2 = 29

b) OPT[j] = { max{ OPT[j-1] + vj, OPT[j-2] + vj * vj-1 } (j ≥ 2)
              v1 (j = 1)
              0 (j = 0)

c) ProductSum (int[] v, n)
    if (n == 0) return 0;
    int[] OPT = new int[n+1];
    OPT[0] = 0;
    OPT[1] = v[1];
    for int j = 2 to n
        OPT[j] = max (OPT[j-1] + v[j], OPT[j-2] + v[j] * v[j-1]);
    return OPT[n];
```

2 최적 하위 구조를 사용할 수 있다.

총 손루선 수를 세기 위해 세트 손루선들을 2개로 나눈다.
첫번째 세트는 m번째 등전(또는 S_m)을 포함하지 않는 용액이고,
두번째 세트는 하나 이상의 S_m을 포함하는 손루선이다.
count(S[], m, n)을 손루선 개수를 세는 함수로 둔 다음 count(S[], m-1, n)과 count(S[], m, n-S_m)의 합으로 쓸 수 있다.
따라서 이 문제는 하위 문제 해결책을 사용하여 풀 수 있기 때문에 최적 하위 구조로 해결 가능하다.

3.

4. ApplesMaximum(A)

```
n = A.length
m = A[0].length
dp[i][j] = A[i][j]
for i = 1 to n do
    for j = 1 to m do
        dp[i][j] = 0
        if (i-1) >= 1 then
            dp[i][j] = max(dp[i][j], A[i] + dp[i-1][j])
        if (j-1) >= 1 then
            dp[i][j] = max(dp[i][j], A[i] + dp[i][j-1])
return dp[n][m]
```

5.

	Total distance
6. A $\xrightarrow{4}$ E	4
A $\xrightarrow{5}$ B	5
A $\xrightarrow{4}$ E $\xrightarrow{11}$ F	15
A $\xrightarrow{5}$ B $\xrightarrow{10}$ G	15
A $\xrightarrow{5}$ B $\xrightarrow{6}$ C	11
A $\xrightarrow{5}$ B $\xrightarrow{6}$ C $\xrightarrow{1}$ D	12
A $\xrightarrow{5}$ B $\xrightarrow{6}$ C $\xrightarrow{1}$ D $\xrightarrow{12}$ H	24

7.