# 2660

# Problem

## 同类题目

# 分类

穷举法

动态规划

# 分析

# 解法1[穷举法]

## 耗时

0~15ms

## 时间复杂度

O()

## 空间复杂度

O(k)

## 分析

直接穷举，寻找最优

## [源码](4-solution/solution_1.cc)

# 解法2[DFS]

## 耗时

## 时间复杂度

O()

## 空间复杂度

O(k)

## 分析



max\_value;

max\_weight;

total\_weight;

total\_value;

dfs(int index, int k, int total\_weight, int total\_value)

{

for (int i = index + 1; i <= **N-k+1**; ++i)

{

total\_weight += current\_weight;

total\_value += current\_value;

**if (1 == k)**

{

**// 已到达叶子节点，停止下潜**

if (total\_weight > max\_weight)

return;

max\_value = max(max\_value, total\_value);

}

else

{

**// 还没有到叶子节点，继续下潜**

dfs(I, k – 1, total\_weight, total\_value);

}

}

}

# 对比

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **描述** | **时间复杂度** | **空间复杂度** | **结果** |
| 解法1 | 穷举组合法 | O() | O(k) | AC，15ms |
| 解法2 |  |  |  |  |