Subset sum problem with DP

This slide is based on this page.
Detail is also at the last of this slide.

Subset sum problem

- Part1: Typical
- Part2: Unbounded
- Part3: With the limited total number of elements
- Part4: With the limited number of each element

Part1

Given set $\{a_1, ..., a_n\}$,

(1)Determine if there is a subset of them with sum equal to K.
(2)Count subsets of them with sum equal to K.
(3)Minimize the size of subset of them with sum equal to K.
(If it doesn't exist, answer INF.)

I think a_i as 0-indexed: $a[0], a[1], \ldots, a[n-1]$.

- $1 \leq n \leq 10^2$
- $1 \leq a[i] \leq 10^3$
- $1 \leq K \leq 10^4$

Part1 (1) 1/2

Determine if there is a subset of them with sum equal to K.

- Def.
  $dp[i][j] = \text{whether } j \text{ can be made with } a[0], \ldots, a[i-1]$
- Init.
  We can make 0 with $\emptyset$:
  - $dp[0][0] = \text{true}$
  - $\text{others} = \text{false}$

Part1 (1) 2/2

- Trans.
  If j is made with $a[0], \ldots, a[i-1]$,
  j is made not using $a[i]$, and $j+a[i]$ is made using $a[i]$
    - $dp[i+1][j] = src \lor dp[i][j]$
    - $dp[i+1][j+a[i]] = src \lor dp[i][j]$
- Ans.
  $dp[n][K]$

Part1 (2) 1/2

Count subsets of them with sum equal to K.

- Def.
  $dp[i][j] = \text{the number of way to make } j \text{ with } a[0], \ldots, a[i-1]$
- Init.
  We can make 0 with $\emptyset$ in one way:
  - $dp[0][0] = 1$
  - $\text{others} = 0$

Part1 (2) 2/2

- Trans.

  If j is made with $a[0], \ldots, a[i-1]$,

  j is made not using $a[i]$, and $j + a[i]$ is made using $a[i]$

  - $dp[i+1][j] = src + dp[i][j]$
  - $dp[i+1][j+a[i]] += src + dp[i][j]$

- Ans.

  $dp[n][K]$

Part1 (3) 1/2

Minimize the size of subset of them with sum equal to K.

(If it doesn't exist, answer INF.)

- Def.

$$dp[i][j] = \text{the minumum size of subset of } \{a[0], \ldots, a[i-1]\} \text{ with sum equal to } j$$

- Init.

  We can make 0 with $\emptyset$ not using any elements:
  - $dp[0][0] = 0$
  - $\text{others} = \text{INF}$

Part1 (3) 2/2

- Trans.

  If j is made with $a[0], \ldots, a[i-1]$,

  j is made not using $a[i]$, and $j + a[i]$ is made using $a[i]$

  - $dp[i+1][j] = \min(src, dp[i][j])$
  - $dp[i+1][j+a[i]] = \min(src, dp[i][j]+1)$

- Ans.

  $dp[n][K]$

Part2

(Same as coin-making problem)

Given a set $\{a_1, a_2, \cdots, a_n\}$.
Each elements can be chosen any times, and **multiset** is a made of them.

> e.g. $\{a, a, b\}$ is not a set, but a multiset because of multiplicity of a.

(1)Determine if there is a multiset of them with sum equal to K.
(2)Count multiset of them with sum equal to K.
(3)Minimize the size of multiset of them with sum equal to K.
(If it doesn't exist, answer INF.)

- $1 \leq n \leq 10^2$
- $1 \leq a[i] \leq 10^3$
- $1 \leq K \leq 10^4$

Part2 (1) 1/3

Determine if there is a multiset of them with sum equal to K.

- Def.
  $dp[i][j] = \text{whether } j \text{ can be made with } a[0], \ldots, a[i-1]$
- Init.
  We can make 0 with $\emptyset$:
  - $dp[0][0] = \text{true}$
  - $\text{others} = \text{false}$

Part2 (1) 2/3

- $\text{Trans(naive)}.$
  If j is made with $a[0], \ldots, a[i-1]$,
  for each $k(0 \leq j + k \cdot a[i] \leq K)$, $(j + k \cdot a[i])$ can be made:
  - $dp[i+1][j + k \cdot a[i]] = src \lor dp[i][j]$

  But time complexity is $O(NK^2)$

Part2 (1) 3/3

- $\mathrm{Trans(good).}$
  If j is made with $a[0], \ldots, a[i-1]$,
  j is made not using $a[i]$, and $j + a[i]$ is made using $a[i]$ one times, or more than two times:
  - $dp[i+1][j] = src \vee dp[i][j]$
  - $dp[i+1][j+a[i]] = src \vee dp[i][j]$
  - $dp[i+1][j+a[i]] = src \vee dp[i+1][j]$
- $\mathrm{Ans.}$
  $dp[n][K]$

Supplement: Code

```
for (int i = 0; i < n; i++) {
  for (int j = 0; j <= K; j++) {
    dp[i + 1][j] |= dp[i][j];
    dp[i + 1][j + a[i]] |= dp[i][j];
    dp[i + 1][j + a[i]] |= dp[i + 1][j];
  }
}
```

If you take care of the order of transition, you can reduce the procedure.

```
for (int i = 0; i < n; i++) {
  for (int j = 0; j <= K; j++) {
    dp[i + 1][j] |= dp[i][j];
    dp[i + 1][j + a[i]] |= dp[i + 1][j];
  }
}
```

Part2 (2)

Count multiset of them with sum equal to K.

- I am too tired to solve it.

Part2 (3)

Minimize the size of multiset of them with sum equal to K.
(If it doesn't exist, answer INF.)

- I am too tired to solve it.

I'm so sorry.

Part3

Given a set $\{a_1, a_2, \cdots, a_n\}$.

You make a subset of them, but **its size must be at most** M

(1)Determine if there is a subset of them with sum equal to K.

(2)Count subset of them with sum equal to K.

(3)Minimize the size of subset of them with sum equal to K.

(If it doesn't exist, answer INF.)

- $1 \leq M \leq n \leq 10^2$
- $1 \leq a[i] \leq 10^3$
- $1 \leq K \leq 10^4$

Part3(1) 1/5

Determine if there is a multiset of them with sum equal to K.

Naive solution

- Def.

$$dp[i][j][k] = \text{whether } j \text{ can be made with } a[0], \ldots, a[i-1], \text{ when the subset size is } k$$

- Init.

  We can make 0 with $\emptyset$:
  - $dp[0][0][0] = \text{true}$
  - $\text{others} = \text{false}$

Part3(1) 2/5

- Trans.
  If j is made with $a[0], \ldots, a[i-1]$,
  j is made not using $a[i]$, and $j + a[i]$ is made using $a[i]$
  - $dp[i+1][j][k] = src \lor dp[i][j][k]$
  - $dp[i+1][j+a[i]][k+1] = src \lor dp[i][j][k]$
- Ans.
  $dp[n][K][0] \lor dp[n][K][1] \lor \cdots \lor dp[n][K][M]$

But the time complexity is $O(nKM)$

Part3(1) 3/5

Good solution

In generary, dp table with bool value is not efficient.
We should consider whether not only true or false, but also more infomation can be gotten from dp table.

Part3(1) 4/5

Infact, it is quite same as Part1(3) except $\mathrm{Ans.}$

- $\mathrm{Def.}$

$dp[i][j] = \text{the minumum size of multiset of } \{a[0], \ldots, a[i-1]\} \text{ with sum equal to } j$

- $\mathrm{Init.}$

  We can make 0 with $\emptyset$ not using any elements:
  - $dp[0][0] = 0$
  - $\text{others} = \mathrm{INF}$

Part3(1) 5/5

- Trans.

  If j is made with $a[0], \ldots, a[i-1]$,

  j is made not using $a[i]$, and $j + a[i]$ is made using $a[i]$

  - $dp[i+1][j] = \min(src, dp[i][j])$
  - $dp[i+1][j+a[i]] = \min(src, dp[i][j]+1)$

- Ans.

  whether $dp[n][K] \leq M$

The time complexity is $O(nK)$

Part3(2) 1/2

Count subset of them with sum equal to K.

Maybe there is no way whose time complexity is $O(nK)$ (I'm not sure.)

- Def.

$dp[i][j][k] = \text{the number of way to make } j \text{ with } a[0], \ldots, a[i-1] \text{ when the subset size is } k$

- Init.

  We can make 0 with $\emptyset$ in one way:

  - $dp[0][0][0] = 1$
  - $\text{others} = 0$

Part3(2) 2/2

- Trans.
  If j is made with $a[0], \ldots, a[i-1]$,
  j is made not using $a[i]$, and $j + a[i]$ is made using $a[i]$
  - $dp[i+1][j][k] = src + dp[i][j][k]$
  - $dp[i+1][j+a[i]][k+1] = src + dp[i][j][k]$
- Ans.
  $dp[n][K][0] + \cdots + dp[n][K][M]$

(3)

Minimize the size of subset of them with sum equal to K.
(If it doesn't exist, answer INF.)

Almost the same as (1).

Part4

Given a set $\{a_1, a_2, \cdots, a_n\}$.
i**-th elements can be chosen** m_i **times**, and **multiset** is a made of them.

(1)Determine if there is a multiset of them with sum equal to K.
(2)Count multiset of them with sum equal to K.
(3)Minimize the size of multiset of them with sum equal to K.
(If it doesn't exist, answer INF.)

- $1 \leq n \leq 10^2$
- $1 \leq a[i], m[i] \leq 10^3$
- $1 \leq K \leq 10^4$

Part4(1) 1/5

Determine if there is a multiset of them with sum equal to K.

Naive solution

- Def.
  $dp[i][j] = \text{whether } j \text{ can be made with } a[0], \ldots, a[i-1]$
- Init.
  We can make 0 with $\emptyset$:
  - $dp[0][0] = \text{true}$
  - $\text{others} = \text{false}$

Part4(1) 2/5

- Trans.
  If j is made with $a[0], \ldots, a[i-1]$,
  for each $k = 0, 1, \ldots, m_i$, $(j + k \cdot a[i])$ can be made:
  - $dp[i+1][j+k \cdot a[i]] = src \lor dp[i][j]$
- Ans.
  $dp[n][K]$

But time complexity is $O(NK \max_i \{m_i\})$

Part4(1) 3/5

Good solution

As Part3(1), dp table with bool value is not efficient.
We should consider whether not only true or false, but also more infomation can be gotten from dp table.

Part4(1) 4/5

- Def.

$$dp[i][j] = \text{maximum remainder of } a[i-1] \text{ to make } j \text{ with } a[0], \ldots, a[i-1]$$

  If j cannot be made, $dp[i][j] := -1.$

- Init.

  We can make 0 with $\emptyset$:
  - $dp[0][0] = 0$
  - $\text{others} = -1$

Part4(1) 5/5

- Trans.

  If j is made with $a[0], \ldots, a[i-1]$,

  j is made not using $a[i]$ (i.e. reminder is still m_i), and $j + a[i]$ is made using $a[i]$

  one times, or more than two times:

  - $dp[i+1][j] = \max(src, m[i])$ if $dp[i][j] \geq 0$
  - $dp[i+1][j+a[i]] = \max(src, m[i]-1)$ if $dp[i][j] \geq 0$
  - $dp[i+1][j+a[i]] = \max(src, dp[i+1][j]-1)$

- Ans.

  whether $dp[n][K] \geq 0$

Part4(2)

Count multiset of them with sum equal to K.

- I'm too tired to solve it.

Part4(3)

Minimize the size of multiset of them with sum equal to K.
(If it doesn't exist, answer INF.)

- I'm too tired to solve it.

I'm so sorry.

Thank you very much

Many problems is based on the page:
典型的な DP (動的計画法) のパターンを整理 Part 1 ～ ナップサック DP 編 ～
(https://qiita.com/drken/items/a5e6fe22863b7992efdb)