

Dynamic Planning

Knapsack

[例題](#) | [参考](#)

n 個の品物があり, i 番目の品物のそれぞれ重さと価値が $w[i]$ 、 $v[i]$ となっている. これらの品物から重さの総和が W を超えないように選んだときの、価値の総和の最大値を求める.

方針

$dp[i+1][j]$ >>> i 番目までの品物の中で重さの総和が j 以下となるように選んだときの最大価値

漸化式

```
if j >= w[i] then
  dp[i+1][j] = max{ dp[i][j - w[i]] + v[i], dp[i][j] }
else
  dp[i+1][j] = dp[i][j]
```

初期条件

```
REP(j, 0, W)
  dp[0][j] = 0
```

Number Partitioning 1

[例題](#) | [参考](#)

n 個の正の整数 $a[0], a[1], \dots, a[n-1]$ の部分和が整数 A になるものはあるか.

方針

$dp[i+1][j]$ >>> i 番目までの整数の中からいくつか選んで総和を j とすることが可能かどうか

漸化式

```

if j >= a[i] then
    dp[i+1][j] = dp[i][j-a[i]] | dp[i][j]
else
    dp[i+1][j] = dp[i][j]

```

初期条件

```

REP(j, 0, A)
    dp[0][j] = (j==0) ? True : False

```

Number Partitioning 2 (数え上げ)

[例題](#) | [参考](#)

n 個の正の整数 $a[0], a[1], \dots, a[n-1]$ から何個か選んで、総和を整数 A にする方法は何通りあるか。(答えは mod $1e9+7$)

方針

```

dp[i+1][j]  >>>   i番目までの整数からいくつか選んで総和をjとする場合の数

```

漸化式

```

if j >= a[i] then
    dp[i+1][j] = dp[i][j - a[i]] + dp[i][j]
else
    dp[i+1][j] = dp[i][j]

```

初期条件

```

REP(j, 0, A)
    dp[0][j] = (j==0) ? 1 : 0;

```