

## DP - Coin Change Problem (A17)

Jika diberikan koin dengan denominasi  $c_1, c_2, \dots, c_k$ , berapa jumlah koin minimal untuk membayar kembalian sebesar  $n$ ? Pada materi tentang Greedy Algorithm kita sudah melihat salah satu versinya, dimana denominasi koin adalah 1, 5, 10, 25, dan pada kasus tersebut dapat diselesaikan dengan cara greedy.

Kita juga telah melihat bahwa jika denominasi diganti menjadi 1, 10, 12, cara greedy tidak lagi berhasil! Salah satu cara untuk menyelesaikan problem ini tentunya dengan brute force, tapi ada cara yang lebih cepat dan elegan, yaitu dengan DP. Berikut ini solusi rekursifnya :

- jika ada koin yang nilainya persis sama dengan  $n$ , maka  $\text{change}[n] = 1$
- jika tidak ada, cari nilai minimum dari  $(1 + \text{change}[n - c_i])$  untuk semua  $c_i < n$

Implementasikanlah algoritma ini dengan cara memoization atau bottom-up untuk menyelesaikan problem dengan input-output seperti berikut :

### Spesifikasi Input

Input diawali dengan dua bilangan bulat  $n \leq 1,000$  yang menunjukkan jumlah kembalian dan  $k \leq 300$  yang menunjukkan jumlah jenis koin. Baris berikutnya berisi  $k$  bilangan bulat yang dipisahkan spasi. Pasti ada koin bernilai 1, dan deretan koin sudah terurut membesar. Masing-masing koin bernilai  $\leq n$ .

### Spesifikasi Output

Output terdiri dari sebuah bilangan bulat yang menunjukkan jumlah koin minimal yang diperlukan untuk membayar kembalian sebesar  $n$ .

### Contoh Input

```
30 3
1 10 12
```

### Contoh Output

```
3
```