

Esercizio 13 PD

Vogliamo realizzare un algoritmo in PD che ci permetta di calcolare il minor numero di monete da impiegare per raggiungere una determinata somma V , con la condizione che abbiamo un numero illimitato di monete $v[i]$, e che le monete siano ordinate $v[1] > v[2] > \dots$.

Praticamente vogliamo minimizzare il numero di monete

$$x_1 + x_2 + \dots + x_n \quad \text{sotto le condizioni che} \quad \sum_{i=1}^n x_i v[i] = V$$

ALGORITMO

~~usando tabelle e fast~~

~~CAMBIO MONETE~~

~~return~~

~~if e(i,j) non è definito~~

$$e(i,j) = \min \{ e(i+1,j), 1 + e(i,j - v[i]) \}$$

~~return e(i,j)~~

con condizione $e(m,j) = j \quad \forall j = 0 \dots V$

CAMBIO MONETE ($v[i \dots m], j$)

if $i = m$ return j

if $e(i,j)$ non è definito

$$e(i,j) = \min \{ e(v[i+1] \dots m, j), 1 + e(v[i \dots m], j - v[i]) \}$$

return $e(i,j)$