

# Hateville

Wednesday, 15 November 2023

16:26

- Noi siamo un ente beneficiario, ed in un quartiere ogni casa ci fa una donazione sse le sue case adiacenti non la fanno. Dobbiamo cercare di massimizzare i fondi
- Esempio:

Casa	Donazione
1	2
2	10
3	5
4	2
5	12
6	1

$S' = \text{Risultato} = \{2, 5\}$

- Definizione del problema:

- o Istanza

Definiamo  $X_n = \{1, \dots, n\}$  che rappresenta il nostro quartiere

$E A \subset X_n$

E rappresentiamo anche  $d_0 \dots d_n$  un vettore che ci dice la quantità di denaro che ogni casa ci dà

Definiamo la seguente funzione:

- $Comp(A)$ : Diciamo che un sottoinsieme  $A$  è compatibile sse p rispettata la seguente regole:

$$\forall i \in A, i - 1 \notin A \wedge i + 1 \notin A \mid \rightarrow Alessio^*$$

$$Comp = \begin{cases} true & Alessio^* \\ false & else \end{cases}$$

- $D(A)$ : Quanto denaro è stato raccolto

$$D = \begin{cases} \sum_{i \in A} d_i & A \neq \epsilon \\ 0 & else \end{cases}$$

- o Soluzione

Dato  $n \in N$  noi definiamo

$$X_n = \{1, \dots, n\}, d_i \forall i \in X_n \mid \rightarrow Istanza$$

Noi dobbiamo determinare un sottoinsieme  $S$  tale

$$COMP(S) = True \wedge D(S) = \max_{\substack{A \subset X_n \\ COMP(A)=True}} \{D(A)\} \rightarrow \text{Soluzione}$$

Comprendiamo velocemente che il sottoproblema è definito in  $i \in \{0, \dots, n\}$

Definiamo una coppia di variabili:

- $OPT_i = D(S_i)$ : Il valore del sottoinsieme che massimizza
- $S_i$ : Il sottoinsieme di  $\{1, \dots, i\}$  che massimizza  $OPT_i$
- Equazione di ricorrenza
  - Caso base:
    - $i = 0$   
 $OPT_i = 0, S_i = \epsilon$
    - $i=1$   
Se abbiamo 1 sola persona, non c'è il problema dei vicini  
 $OPT_i = d_i, S_i = \{i\}$
  - Pass ricorsivo  
Qui abbiamo 2 casi:
    - Accettiamo  
Se lo accettiamo, allora non possiamo guardare il vicino ma il vicino del vicino, quindi  $i-2$   
 $OPT_i = OPT_{i-2} + d_i, S_i = S_{i-2} \cup \{i\}$
    - Non accettiamo  
Quindi il problema andrà al prossimo vicino  
 $OPT_i = OPT_{i-1}, S_i = S_{i-1}$

Ed ovviamente prendiamo il massimo dei due

Riscriviamola bene:

$$S_i = \begin{cases} 0 & i = 0 \\ d_i & i = 1 \\ S_{i-1} & OPT_{i-1} \geq OPT_{i-2} + d_i \\ S_{i-2} \cup \{i\} & else \end{cases}$$

- Pseudocodice
  - Ricorsivo  
Def HatevilleRic(i):  
If  $i=0$ :  
Return (0, {})  
Else:  
(OPT1, S1) = HatevilleRic(i-1)  
(OPT2, S2) = HatevilleRic(i-2)  
OPT2 = OPT +  $d_i$   
S2 = S2  $\cup$  {i}  
If  $OPT1 \geq OPT2$ :  
Return (OPT1, S1)  
Else:

Return (OPT2, S2)

- Bottomup

Def Hateville(n):

OPT[0] = 0

S[0] = 0

For i=1 to n:

OPT1 = OPT[i-1]

OPT2 = OPT[i-2] + d<sub>i</sub>

If OPT1 ≥ OPT2:

S[i] = S[i-1]

OPT[i] = OPT1

Else:

S[i] = S[i-2] ∪ {i}

OPT[i] = OPT2

Return (OPT[n], S[n])