

Seminar 2

Stefan Popescu

April 2023

Vouchere de Vacanță

(10p) George a intrat în concediu, și ca atare dorește să consume cât mai mult din pachetul său de vouchere de vacanță. El și-a făcut o listă cu n locații unde poate cheltui din suma alocată. Știind că George nu are de gând să cheltuiască din banii proprii decât pe transportul între locații și că el se poate deplasa între oricare două locații, ajutați-l pe George să găsească o listă de locații pe care să le viziteze astfel încât să își folosească voucherele cât mai eficient.

Cerințe:

Se cere un algoritm 1/2-aproximativ care rezolvă problema lui George în timp eficient din punct de vedere al complexității timp. (pseudocod + justificare)

Algoritmul va avea ca output lista de locații și suma totală ce va fi cheltuită.

Notații și indicații:

- Notăm cu S suma de bani pe care o are George sub formă de vouchere.
- Locațiile sunt etichetate cu numerele $1, 2, \dots, n$. Costurile celor n locații sunt notate cu C_1, C_2, \dots, C_n
- Suma de bani S este alocată pe un card, nu sub formă de tichete. Din cont se poate extrage orice sumă disponibilă.
- Se lucrează cu numere întregi.

Camioane și Transporturi

(15p) Avem n colete de transportat, fiecare având greutatea de w_1, w_2, \dots, w_n . Pentru a le transporta, putem folosi un număr de camioane, fiecare având capacitatea de transport G . presupunem că $w \leq G$, pentru orice i . Ne dorim să minimizăm numărul de camioane folosite. Considerăm următorul plan de încărcare a camioanelor:

Odată ce avem la dispoziție un camion pentru a fi încărcat, iterăm prin mulțimea coletelor, încărcându-le în camion, până când dăm peste primul colet ce nu mai încapă. În acel moment considerat că am terminat de încărcat camionul curent și trecem la următorul camion, prima dată încărcând coletul care nu a mai încăput în cel precedent.

Cerințe:

- Arătați, printr-un exemplu simplu, că metoda de mai sus nu furnizează soluția optimă. **(5p)**
- Arătați totuși că soluția de mai sus este un algoritm 2-aproximativ pentru problema noastră. **(10p)**

Notații:

n - numărul de colete

G - capacitatea unui camion

W - Lista cu greutatea coletelor în ordinea în care sosesc spre a fi procesate.

OPT – soluția optimă care există: Numărul minim de camioane ce poate fi folosit pentru a transporta cele n colete de greutate totală G

ALG – soluția oferită de algoritmul nostru: numărul de camioane folosite conform algoritmului

$G[i]$ - greutatea cu care este încărcat camionul i în configurația dată de algoritmul nostru. Evident $G[i] \leq G$.

Cadourile lui Moș Crăciun

(30p) O clasă de elevi este formată din m copii. La această clasă Moș Crăciun aduce n cadouri, fiecare dintre ele având o valoare notată $val_1, val_2, \dots, val_n$. Moșul, grăbit fiind, lasă task-ul împărțirii cadourilor pe umerii profesorului de informatică. Acesta își alege următorul obiectiv pentru împărțirea cadourilor: să maximizeze valoarea cadourilor primite de către copilul care primește cel mai puțin. (Altfel spus, copilul cel mai “vitregit” în urma împărțirii să primească totuși cadouri de o valoare cât mai bună / Să existe un “spread” cât mai bun al cadourilor). În timp ce se gândește la o soluție de implementare, profesorul își amintește de la cursurile de algoritmică din facultate că astfel de probleme de optim sunt NP-hard! Fiind cuprins de groază (dar și de o nostalgie pentru cursurile de algoritmică din facultate) el vă cere vouă ajutorul.

Notății:

OPT – soluția optimă care există: Valoarea totală a cadourilor primite de copilul cel mai “vitregit” în o configurație optimă

ALG – soluția oferită de algoritmul vostru: Valoarea totală a cadourilor primite de copilul cel mai “vitregit” în urma algoritmului vostru

$C[k]$ - lista cadourilor primite de către copilul k la finalul algoritmului vostru

$W(K)$ – valoarea totală a cadourilor primite de către copilul k la finalul algoritmului vostru

$val(i)$ – valoarea cadoului i

$Val = \sum_{1 \leq i \leq n} val(i)$ – valoarea totală a cadourilor

Copiii vor fi indexați folosind variabile de forma k, q, k', q' , etc... Cadourile vor fi indexate folosind variabile de forma i, j, i', j' , etc...

Restricții: Considerăm că $n \geq m$ și că $val(i) \leq \frac{Val}{2m}$ pentru oricare cadou i .

Cerințe:

- Descrieți un algoritm $\frac{1}{2}$ aproximativ pentru problema cadourilor în complexitate $\mathcal{O}(n \log m)$ **(10p)**
- Fie k acel copil pentru care $ALG = W(K)$. Fie i ultimul cadou primit de un copil oarecare q ($q \neq k$). Care este relația între $W(K)$ și $W(Q) - val(i)$? Justificați. **(5p)**
- Pe baza punctului b) arătați că $ALG \geq \frac{Val}{2m}$ **(5p)**
- Demonstrați că algoritmul descris la punctul a) este $\frac{1}{2}$ aproximativ **(5p)**
- Dați un exemplu format din minimum 2 copii și 4 cadouri pentru care algoritmul vostru nu găsește soluția optimă. Spuneți care este soluția optimă. Spuneți care este soluția dată de algoritmul vostru. **(5p)**