Fie L – lista obiectelor sortate după raportul valoare/greutate

Fie O_p – obiectul cu prof itul cel mai mare din lista de obiecte.

S=0, $G=capacitatea\ rucsacului$;

Pentru f iecare O:L

Dacă greutate(
$$O$$
) $\leq G$, atunci $S + = val(O)$, $G - = greutate(O)$

$$ALG(I) = max(S, O_p)$$

În primul rând, este evident că algoritmul de mai sus ne oferă o soluție fezabilă. Elementele care au ca suma valorilor S vor avea o greutate totală ≤ capacitatea rucsacului, respectiv O_p încape și el în rucsac de unul singur.

Trebuie să justificăm doar factorul de aproximare.

Fie OPT_{1/0} valoarea optima pentru Problema Rucsacului in varianta 1/0, respectiv OPT_G valoarea optimă, furnizată de algoritmul de tip greedy pentru problema rucsacului in varianta în care aveam voie să "tăiem" obiecte pentru a le încărca în rucsac.

Cum este $OPT_{1/0}$ față de OPT_G ? $OPT_{1/0}$ <= OPT_G

$$OPT_{1/0} \le OPT_G$$

Fie k indicele primului obiect care nu este adaugat in algoritmul de la inceputul paginii.

$$OPT_{1/0} \leq OPT_G \leq \sum_{1 \leq i \leq k} val(O_i) = \sum_{1 \leq i < k} val(O_i) + val(O_k) \leq \sum_{1 \leq i < k} val(O_i) + val(O_p)$$

$$ALG = max(S, O_p)$$

$$OPT_{1/0} \leq \sum_{1 \leq i < k} val(O_i) + val(O_p) \leq ALG + ALG = 2 \cdot ALG$$

$$OPT_{1/0} \le 2 \cdot ALG$$

Ex intrare pt care abaterea e maxima

G=100

Ob (val/greutate)=[(50+eps1)/(50+eps2),50/50,50/50]

cu eps1>eps2>0

Evident profitul maxim este 100

profitul solutiei algoritmului este 50+eps1

$$ALG(I) \cong \frac{1}{2} \cdot OPT(I)$$

deci 1/2 este un "tight upper bound"