

Student: Chirculete Vlad Mihai

Grupa: 231

## **Tema1**

### **Varianta 2, problema 3**

**Corectitudine:** Presupunem ca avem activitatile sortate descrescator in functie de profit. Initializam secventa rezultat cu prima activitate obtinuta in urma sortarii. Parcurgem toate celelalte activitati ramase si vom efectua urmatoarele operatii:

- verificam daca activitatea la care s-a ajuns in parcurgere poate fi introdusa in secventa rezultat (se poate efectua fara a rata deadlineul), in caz afirmativ o introducem in secventa rezultat.
- daca activitatea curenta nu se poate efectua fara a rata deadlineul atunci aceasta este ignorata.

Aceasta solutie Greedy consta in asocierea de sloturi libere cate unei activitati. Parcurgem toate sloturile posibile pentru o activitate si ii asociem acesteia cel mai mare slot de timp posibil (dar care este mai mic decat deadlineul) care a fost gasit liber.

#### **Ce reprezinta cel mai mare slot de timp?**

Sa presupunem ca o activitate A1 are deadlineul  $t=5$ . Asociem cel mai mare slot liber acestei activitati (dar mai mic decat deadlineul) ex: slotul 4-5 pentru aceasta activitate. Acum o alta activitate A2 cu deadlineul egal cu 5 este gasita, asadar slotul de timp alocat acesteia va fi 3-4 deoarece 4-5 este deja alocat activitatii A1.

#### **Este optim sa alocam unei activitati cel mai mare slot de timp liber?**

Alocam cel mai mare slot de timp disponibil unei activitati deoarece daca am alocat un slot mai mic acesteia decat cel disponibil s-ar putea ca mai tarziu sa gasim o alta activitate care nu va putea fi introdusa deoarece si-ar rata deadlineul.

De exemplu:

A1 cu  $t_1 = 5$ , profit 40

A2 cu  $t_2 = 1$ , profit 20

Sa presupunem ca pentru A1 am alocat slotul 0-1. Acum A2 nu mai poate fi desfasurata deoarece vom desfasura activitatea A1 in acel slot de timp, astfel obtinand un profit de 40 cand am fi putut obtine un profit de 60 daca am fi alocat primul slot lui A2 si apoi un alt slot lui A1.

**Asadar este optim sa alocam unei activitati cel mai mare slot de timp liber si nu unul mai mic astfel existand posibilitatea de a desfasura cat mai multe activitati posibile.**

#### **Solutia propusa cu structuri de date de tipul "Disjoint Set":**

La inceput toate sloturile de timp sunt seturi individuale. Mai intai gasim deadlineul maxim al activitatilor. Fie deadlineul maxim  $m$ . Cream  $m+1$  seturi individuale. Daca o activitate este asociata unui slot de timp  $t$  unde  $t \geq 0$ , atunci activitatea este rezervata in timpul  $[t-1, t]$ . Asadar un set cu valoarea  $X$  reprezinta slotul de timp  $[X-1, X]$ .

Trebuie sa tinem cont de cel mai mare slot de timp disponibil care poate fi alocat unei activitati cu un anumit deadline. Folosim vectorul de tati al structurii de tip “Disjoint Set” pentru acest scop. Radacina arborelui este tot timpul cel mai mare slot disponibil. Daca pentru un deadline **d** nu exista un slot disponibil, atunci radacina este 0.

**Complexitate:**

- atribuirii simple
- **for** pentru citirea din fisier, n iteratii (complexitate:  **$O(n)$** )
- **for** pentru gasirea deadlineului maxim, n iteratii (complexitate:  **$O(n)$** )
- **for** pentru scrierea in fisier (complexitate:  **$O(n)$** )
- **find** si **merge** in “Disjoint data set” (complexitate:  **$O(\alpha(n))$** )

**Total:  $O(\alpha(n))$**