Proiectarea algoritmilor
Lucrare de laborator nr. 12
Paradigma *greedy*
Drumuri minime care pleacă dintr-un vârf i_0 dat, într-un digraf ponderat
Algoritmul lui Dijkstra

Cuprins

1 Algoritmul lui Dijkstra

1.1 Descriere

Determină drumurile minime care pleacă dintr-un vârf i_0 dat, într-un digraf ponderat $(G, \ell) = (\langle V, A \rangle, \ell)$. Ponderile $\ell_{i,j}$ sunt ≥ 0. Pentru fiecare vârf i, $D[i]$ va fi lungimea drumului minim de la i_0 la i şi $P[i]$ va fi predecesorul lui i pe drumul minim de la i_0 la i.

1.2 Pseudocod

Notaţii:

- $(G,\ell) = (\langle V,A\rangle,\ell)$ este un digraf ponderat.
- $D[0..n-1]$ şi $P[0..n-1]$ sunt vectori de dimensiune n.
- $L[0..n-1,0..n-1]$ este un tablou bidimensional de marime nxn.
- S este mulţimea vârfurilor selectate; iniţial $S = \emptyset$.

Premise:

- Inițial, $L[i,j] = \begin{cases} \ell_{i,j}, & \text{dacă } (i,j) \in A \\ 0, & \text{dacă } i = j \\ \infty, & \text{altfel.} \end{cases}$

```
procedure Dijkstra(G,L,i0,D,P)
   for i ← 0 to n-1 do
       P[i] = i0
       D[i] = ℓ_{i0,i}
   S = {i0}
   while S ≠ V
       i = k  pentru  care  D[k] = min{D[j]; j ∈ V\S}
       S = S ∪ {k}
       for fiecare  j ∈ listaDeAdiacenta(i)  și  j ∈ V\S
           if (D[j] >D[i]+L[i,j])
              D[j] = D[i] + L[i,j]
              P[j] = i
```

2 Sarcini de lucru

Scrieţi un program C/C++ care implementează algoritmul lui Dijkstra.

Bibliografie

[1] Lucanu, D. şi Craus, M., *Proiectarea algoritmilor*, Editura Polirom, 2008.

[2] Moret, B.M.E.şi Shapiro, H.D. , *Algorithms from P to NP: Design and Efficiency*, The Benjamin/Cummings Publishing Company, Inc., 1991.