

Structuri de date și algoritmi

Proiect 4: *TAD MultiDicționar (MultiMap) - implementare folosind listă simplu înlănțuită dinamic*

Nume: Bădiță Marin-Georgian
Grupa: 211

1. Enunțul problemei

În capitala Azerbaidjanului, sătul fiind de traficul infernal, primarul dorește să îmbunătățească rețeaua de transport. Având la dispoziție intersecțiile între care există drumuri, precum și numărul total de vehicule care pot trece prin fiecare drum la un moment dat, primarul dorește să afle:

- a) dacă toate mașinile care intră în oraș pot să și iasă
- b) dacă rețeaua pe care o are la dispoziție este destul de eficientă astfel încât o mașină să poată parcurge toate drumurile din oraș trecând o singură dată prin fiecare.
- c) capacitatea unui drum nou adăugat astfel încât toate mașinile care intră în oraș să poată ieși
- d) numărul minim de drumuri care trebuie adăugate astfel încât să existe un drum cu proprietatea de la b)

Rezolvarea problemei se reduce, de fapt, la interpretarea intersecțiilor ca noduri într-un graf și a drumurilor ca muchiile dintre acestea așa încât în rezolvarea problemei enunțate se vor folosi algoritmi de flux (Ford Fulkerson) și ciclu/lanț eulerian, algoritmi specifici grafurilor, care pot fi foarte bine reprezentați utilizând un MultiDicționar, cheile acestuia făcând parte din mulțimea nodurilor, iar pentru fiecare cheie (nod) se poate reține lista de vecini a

acestui, păstrând complexitățile operațiilor cât mai bune specifice listelor înlănțuite.

2. Reprezentarea TAD-ului

TPereche:

c: TCheie
v: TValoare

MD:

prim: \uparrow TNod
ultim: \uparrow TNod

TNod

e: TPereche
urm: \uparrow TNod

3. Specificarea și interfața TAD-ului

Un MultiDicționar este un container asociativ care conține elemente formate din perechi de tip $\langle \text{TCheie}, \text{TValoare} \rangle$, o cheie putând avea asociate una sau mai multe valori, înseamnă că este posibil să avem mai multe valori care sunt asociate la aceeași cheie, deci cheile nu sunt întotdeauna unice.

$Md = \{ md \mid md \text{ este un MultiDicționar cu elemente de tip } (c, v) \text{ unde } c \in \text{TCheie} \text{ si } v \in \text{TValoare} \}$

creeaza(md):

descriere: creează un nou MultiDicționar
pre: True
post: $md \in MD$, (md e un MultiDicționar vid)

distrug(md):

descriere: eliberează spațiul alocat pentru md
pre: $md \in MD$
post: md a fost "distrus" (spațiul a fost eliberat)

adauga(md, c, v):

descriere: adaugă la md perechia (c, v)

pre: $md \in MD$, $c \in TCheie$, $v \in TValoare$

post: perechea (c, v) a fost adăugată în md

sterge(md, c, v):

descriere: șterge din md perechea (c, v)

pre: $md \in MD$, $c \in TCheie$, $v \in TValoare$

post: perechea (c, v) a fost ștearsă din md

cauta(md, c, l):

descriere: caută o cheie in MultiDicționar

pre: $md \in MD$, $c \in TCheie$

post: $l \in L$ (listă),

cauta $\leftarrow \begin{cases} \text{true si } l \text{ conține lista valorilor asociate cu } c \\ \text{dacă } c \text{ este cheie în } md \\ \text{fals și } l \text{ este lista vidă, altfel} \end{cases}$

iterator(md, i):

descriere: creează un iterator pe md

pre: $md \in MD$

post: $i \in I$, i este iterator pe md

dim(md):

descriere: calculeaza dimensiunea MultiDicționarului

pre: $md \in MD$

post: dim \leftarrow dimensiunea MultiDicționarului md (numărul de elemente) - număr natural

vid(md):

descriere: verifica daca md este MultiDicționarul vid

pre: $md \in MD$

post: $vid \leftarrow true$, dacă md este MultiDicționarul vid
false, altfel

Iterator Multidicționar

Un iterator este o structură folosită pentru a parcurge elementele unui container, în cazul nostru elementele unui MultiDicționar

$I = \{ i \mid i \text{ este un iterator pe multidicționar cu elemente de forma perechi } (c, v), \text{ cu } c \in TCheie \text{ si } v \in TValoare \}$

IteratorMultiDicționar - reprezentare:

multiDicționar: MD (MultiDicționar)

curent: $\uparrow TNode$

IteratorMultiDicționar - interfață:

creeaza(i, md):

descriere: creează un iterator pe md

pre: $md \in MD$

post: $i \in I$ (se creează un iterator pe md . Elementul curent din iterator referă "prima" pereche din MultiDicționar)

element(i, e):

descriere: extrage în e perechea referită de curent

pre: $i \in I$

post: $e \in TValoare$, e este perechea curentă din iterație
(perechea din MultiDicționar referită de curent)

valid(i):

descriere: verifică dacă iteratorul este valid

pre: $i \in I$

post: valid \leftarrow true, dacă iteratorul referă către un element valid
false, altfel

urmator(i):

descriere: trece la perechea următoare din iterație

pre: $i \in I$

post: curent' referă 'următoarea' pereche din MultiDicționar
față de cea referită de curent