

TAD Lista

Domeniu:

$L = \{ l \mid l \text{ este o lista cu elemente de tip } TElement, \text{ fiecare element avand o pozitie unica in } l \}$

Operatii:

- **creeaza(l)**
pre: true
post: $l \in L, l = \phi$
- **element(l, p, e)**
pre: $l \in L, p \in Natural, valid(p)$
*post: $e \in TElement$,
 $e = \text{elementul de pe pozitia } p \text{ din } l$
@arunca exceptie daca p nu e valida*
- **pozitie(l, e)**
pre: $l \in L, e \in TElement$
*post: $pozitie = p \in Natural$
$$p = \begin{cases} \text{prima pozitie a lui } e \text{ din } l, & \text{daca } e \in l \\ -1, \text{ altfel} \end{cases}$$*
- **modifica(l, p, e)**
*pre: $l \in L, p \in Natural, valid(p)$,
 $e \in TElement$*
*post: elementul de pe pozitia p din l' = e
@ arunca exceptie daca p nu e valida*
- **adaugaInceput(l, e)**
pre: $l \in L, e \in TElement$
post: elementul e a fost adaugat la inceputul listei l
- **adaugaSfarsit(l, e)**
pre: $l \in L, e \in TElement$
post: elementul e a fost adaugat la sfarsitul listei l
- **adaugaDupa(l, p, e)**
*pre: $l \in L, p \in Natural, valid(p)$,
 $e \in TElement$*
post: elementul e a fost inserat in lista l dupa pozitia p,

@arunca exceptie daca p nu e valida
- **adaugaInainte(l, p, e)**
*pre: $l \in L, p \in Natural, valid(p)$,
 $e \in TElement$*
*post: elementul e a fost inserat in lista l inainte de pozitia p,
@ arunca exceptie daca p nu e valida*
- **sterge(l, p, e)**
pre: $l \in L, p \in Natural, valid(p)$
*post: $e \in TElement$, elementul e de pe pozitia p a fost sters din lista l.
@ arunca exceptie daca p nu e valida*
- **cauta(l, e)**
pre: $l \in L, e \in TElement$
post: $cauta = \begin{cases} true, \text{daca } e \text{ a fost gasit in } l \\ fals, \text{altfel} \end{cases}$
- **vida(l)**
pre: $l \in L$
post: $vida = \begin{cases} true, \text{daca } l = \phi \\ false, \text{altfel} \end{cases}$
- **dim(l)**
pre: $l \in L$
*post: $dim = n, n \in Natural$
 $n = \text{numarul de elemente ale listei } l$*
- **distruge(l)**
pre: $l \in L$
post: l a fost 'distrusa' (spatiul de memorie alocat a fost eliberat)
- **iterator(l, i)**
pre: $l \in L$
post: $i \in I$, i este iterator pe lista l

TAD Multidictionar Ordonat

Domeniu:

$MDO = \{ mdo \mid mdo \text{ este un multidictionar ordonat, cu perechi } TCheie, TValoare, \text{ unde pe multimea elementelor } TCheie \text{ putem defini o relatie de ordine, } R \}$

Operatii:

- **creeaza(mdo, R)**
pre: R – relatie pe multimea cheilor
post: $mdo \in MDO, mdo = \phi$
- **distruge(mdo)**
pre: $mdo \in MDO$
post: mdo a fost “distrusa” (spatiul alocat a fost eliberat)
- **adauga(mdo, c, v)**
pre:
 $mdo \in MDO, c \in TCheie, v \in TValoare$
post: perechea $\langle c, v \rangle$ a fost adaugata in mdo
- **sterge(mdo, c, v)**
pre:
 $mdo \in MDO, c \in TCheie, v \in TValoare$
post: perechea $\langle c, v \rangle$ a fost stearsa din mdo
- **cauta(mdo, c, l)**
pre: $mdo \in MDO, c \in TCheie$
post:
 $l \in L,$
$$\rightarrow \begin{cases} true \text{ si } l \text{ este lista valorilor asociate cu } c, \\ \quad \text{daca } c \text{ este cheie in } mdo \\ fals \text{ si } l = \phi, \text{ altfel} \end{cases}$$
- **iterator(mdo, i)**
pre: $mdo \in MDO$
post: $i \in I, i$ este iterator pe mdo

Alte operatii posibile:

- **multimeaCheilor(mdo, m)**
pre: $mdo \in MDO$
post: $m \in M, m$ este multimea tuturor cheilor din mdo
- **multimeaValorilor(mdo, m)**
pre: $mdo \in MDO$
post: $m \in M, m$ este multimea tuturor valorilor din mdo