

Documentatia pentru fisierele IN si OUT

ERRORS	
<i>INVALID_INPUT</i>	daca inputul pentru comanda nu este corect
<i>INVALID_COMMAND</i>	daca comanda nu este valida
<i>INDEX_OUT_OF_BOUNDS</i>	in cazul in care se cere un index, dar acesta depaseste limitele
<i>NOT_FOUND</i>	daca un anumit element nu a fost gasit
<i>CURRENT_STRUCTURE_UNDEFINED</i>	in momentul in care se citeste o comanda care lucreaza pe structura curenta, dar aceasta nu a fost creata
<i>PREVIOUS_STRUCTURE_UNDEFINED</i>	la liste, daca dorim sa facem merge intre lista curenta si cea anterioara, dar cea anterioara nu a fost definita
<i>NUMBER_OF_STRUCTURES_EXCEEDED</i>	goto obs. 5
<i>CAPACITY_LIMIT_REACHED</i>	daca se depasesc 10.000.000 de elemente

	Observatii
1.	O comanda se considera valida daca pe linia sa se afla doar numele comenzii. Orice alt caracter in plus (chiar si spatiu) va rezulta in <i>INVALID_COMMAND</i> . Daca linia este goala, rezultatul va fi <i>INVALID_COMMAND</i> . Exemple de comenzi valide: "VectorRead", "VectorAdd", "ListMerge" Exemple de comenzi invalide: " VectorRead", "VectorRead ", "vectorread"
2.	Inputul comenzilor poate fi separat prin oricate spatii sau taburi
3.	Dupa comenzile fara input nu se va lasa linie libera
4.	Toate argumentele trebuie sa fie de tip int, in caz contrar se va afisa ca output <i>INVALID_INPUT</i>
5.	Orice comanda cu Read va face o noua structura in care va pune inputul. In cazul in care s-a ajuns la ultima structura si se incearca din nou Read, se va afisa ca output <i>NUMBER_OF_STRUCTURES_EXCEEDED</i>
6.	La orice operatie de Add sau Insert daca nu exista structura curenta creata se va afisa <i>CURRENT_STRUCTURE_UNDEFINED</i>
7.	Orice operatie care implica depasirea limitei de 10.000.000 elemente se ignora (nu

	se efectueaza) se va afisa <i>CAPACITY_LIMIT_REACHED</i>
8.	Fisierele .in nu pot fi vide si nu pot sa lipseasca (ex: nu poate exista 007.in daca nu exista toate pana la 006.in inclusiv)
9.	In fisierele .res se va lasa o linie noua la sfarsit, iar dupa comanda Print se va lasa un spatiu suplimentar dupa ultimul element

Nume Comanda	Descriere	Input	Output
VectorGoTo	Seteaza vectorul curent	-un singur argument - <u>indicele noului vector</u> -indicele trebuie sa fie de tip int si cuprins intre 0 si numarul maxim de structuri	<i>INDEX_OUT_OF_BOUNDS</i> <i>INVALID_INPUT</i>
VectorRead	Citeste un vector si il seteaza ca si vector curent	-maxim 10 milioane de argumente - <u>elementele care se vor adauga in vector</u> -cel putin un element	<i>INVALID_INPUT</i> <i>CAPACITY_LIMIT_REACHED</i> <i>daca sunt peste 10.000.000 elemente => Se ignora vectorul citit.</i>
VectorAdd	Adauga un element la vectorului curent	-un singur argument - <u>elementul care se va aduga la sfarsitul vectorului</u>	<i>INVALID_INPUT</i> Daca vectorul curent nu este definit se afiseaza <i>CURRENT_STRUCTURE_UNDEFINED</i> <i>CAPACITY_LIMIT_REACHED</i> <i>vectorul curent are deja 10.000.000 de elemente => nu se adauga</i>
VectorLength	Afiseaza lungimea vectorului curent	-niciun argument	lungimea vectorului curent Daca vectorul curent nu este definit se afiseaza <i>CURRENT_STRUCTURE_UNDEFINED</i>
VectorRemovePosition	Stergerea unui element cu indicele dat (din	-un singur argument - <u>indicele elementului ce urmeaza a fi sters</u>	<i>INDEX_OUT_OF_BOUNDS</i>

	vectorul curent)		INVALID_INPUT
VectorRemoveValue	Stergerea tuturor elementelor cu valoarea data (din vectorul curent)	-un singur argument <u>-valoarea elementului ce va fi sters</u>	INVALID_INPUT NOT_FOUND -daca nu exista niciun element cu valoarea data
VectorSearch	Cauta un element in vectorul curent	-un singur argument <u>-valoarea elementului de cautat</u>	-pozitia primei aparitii a elementului in vector NOT_FOUND INVALID_INPUT
VectorPrint	Afiseaza vectorul complet	-nicun argument	-toate elementele vectorului curent CURRENT_STRUCTURE_UNDEFINED -daca vectorul este gol se afiseaza linie goala
ListGoTo	La fel ca VectorGoTo		
ListAdd	Adauga un element la sfarsitul listei curente	-un singur argument <u>-elementul de adaugat</u>	INVALID INPUT Daca lista curenta nu este definita se afiseaza CURRENT_STRUCTURE_UNDEFINED CAPACITY_LIMIT_REACHED
ListRead	Citeste lista dublu inlantuita si o seteaza ca fiind curenta	-maxim 10 milioane de argumente <u>-elementele care se vor adauga in lista</u> <u>-cel putin un element</u>	INVALID_INPUT CAPACITY_LIMIT_REACHED
ListPrint	Tipareste lista curenta	-niciun argument	Afiseaza toate elementele ultimei liste CURRENT_STRUCTURE_UNDEFINED
ListMerge	Face merge la lista curenta cu lista	-niciun argument	CURRENT_STRUCTURE_UNDEFINED

	<p>anterioara listei curente.</p> <p><u>Cele doua liste trebuie sa fie sortate!</u></p> <p>Lista rezultata se va seta ca si curenta.</p>		<p><i>PREVIOUS_STRUCTURE_UNDEFINED</i></p> <p><i>NUMBER_OF_STRUCTURES_EXCEEDED</i></p> <p><i>CAPACITY_LIMIT_REACHED</i> <i>daca rezultatul are peste 10.000.000 elemente => se ignora rezultatul merge-ului.</i></p>
ListSort	Sortează lista curentă folosind algoritmul <i>merge sort</i>	-niciun argument	<i>CURRENT_STRUCTURE_UNDEFINED</i> <i>daca nu avem nici o lista</i>
HashTableGoTo	- -		
HashTableRead	Citeste un hash table	<p>-maxim 10 milioane de argumente</p> <p><u>-elementele de adaugat</u></p> <p><u>-cel putin un element</u></p>	<p>INVALID_INPUT</p> <p><i>CAPACITY_LIMIT_REACHED</i></p>
HashTableSearch	Cauta intr-un hash table	<p>-un singur argument</p> <p><u>-valoarea elementului de cautat</u></p>	<p>FOUND</p> <p>NOT_FOUND</p> <p>INVALID_INPUT</p>
TreeGoTo	- -		
TreeRead	Citeste un arbore si il seteaza ca arbore curent.	<p>-maxim 10 milioane argumente</p> <p><u>-se furnizeaza elementele separate prin unul sau mai multe spatii (in orice ordine)</u></p> <p><u>-cel putin un element</u></p>	<p>INVALID_INPUT</p> <p><i>CAPACITY_LIMIT_REACHED</i></p>
TreeInsert	Insereaza un element in arbore. Elementul deja existent se ignora.	<p>-un singur argument</p> <p><u>-se furnizeaza elementul de inserat in arbore</u></p>	<p>INVALID_INPUT</p> <p>Daca nu exista tree definit se afiseaza CURRENT_STRUCTURE_UNDEFINED</p> <p><i>CAPACITY_LIMIT_REACHED</i></p>

TreeSearch	Cauta un element in arbore	-un singur argument <u>-se furnizeaza elementul de cautat</u>	FOUND NOT_FOUND INVALID_INPUT
TreeRemove	Elimina un element din arbore	-un singur argument <u>-pe prima linie, se furnizeaza elementul de eliminat</u>	NOT_FOUND INVALID_INPUT
HeapGoTo			
HeapRead	Citeste un heap si il seteaza ca heap curent.	-maxim 10 milioane argumente <u>-se furnizeaza elementele separate prin unul sau mai multe spatii (in orice ordine)</u> <u>-cel putin un element</u>	INVALID_INPUT <i>CAPACITY_LIMIT_REACHED</i>
HeapInsert	element in heap insereaza un .	-un singur argument -se furnizeaza elementul de adaugat	INVALID_INPUT Daca Heapul curent nu este definit se afiseaza CURRENT_STRUCTURE_UNDEFINED <i>CAPACITY_LIMIT_REACHED</i>
HeapRemove	Elimina radacina heap-ului	-niciun argument	-radacina heap-ului NOT_FOUND - daca heap-ul este gol

Nelamuriri

1	Comanda Read suprascrie peste vectorul curent? R:Nu. Se creeaza o noua structura (daca se poate)
2	Daca lista curenta e ultima ce se va intampla dupa ListMerge cu lista rezultata? R: <i>NUMBER_OF_STRUCTURES_EXCEEDED</i>
3	La heap remove unde se elimina radacina. Heapul e min sau max ? R: minheap

4	Putem avea intr-un test mai multe structuri de tipuri diferite ? R:evident (este trivial)
5	Intr-un test putem avea maxim 10 structuri de orice tip sau maxim 10 structuri pentru fiecare tip? R:maxim 10 pt fiecare
6	<i>CURRENT_STRUCTURE_UNDEFINED ?</i> <i>Cand/Cum se defineste o structura ?</i> <i>Doar comenzile Read definesc o structura ?</i> <i>R:Cand faci Read se creeaza o noua structura pe urmatoarea pozitie libera</i> <i>R+: Comanda ListMerge <u>poate</u> defini o noua structura.</i>
7	<i>La comenzile gen VectorAdd 4 5 da parametru invalid sau il adauga pe 4</i> <i>R:INVALID_INPUT</i>
8.	<i>Inputul comenzilor poate fi separat prin TAB-uri ?</i> <i>R:Da</i>
9.	<i>Care este prioritatea erorilor? De exemplu la o comanda Read ce eroare se afiseaza dintre NUMBER_OF_STRUCTURES_EXCEEDED sau CAPACITY_LIMIT_REACHED daca au loc ambele erori ?</i> <i>R: mai logic ar fi number of structures exceeded intai... mai intai verifici daca ai loc, abia apoi citesti. Nu citesti numa ca apoi sa iti dai seama ca tu n-ai unde sa pui toate elementele</i>
10.	<i>New line la sfarsitul fisierului inseamna INVALID_COMMAND?</i> <i>R:daca e la fisierul de input, da</i>
11.	<i>Arborele poate avea elemente duplicate ?</i> <i>R: Nu. Se vor ignora duplicatele.</i>