# **Structuri de date si algoritmi**

Laborator 1-5

* Reprezentarea cozii:

Coada:

cp:intreg reprezentand capacitatea maxima de memorare

inceput: intreg care retine pozitie prin intermediul careia gasim primul element adaugat

sfarsit: intreg care retine pozitia prin intermediul careia gasim ultimul element adaugat

array: este o lista de TElemente cu diminesiunea de [1->cp]

* Subalgoritmi

Subalgoritm adauga (array,e)

//Complexitatea este θ(n) in worst case -trece capcitatea

// Complexitatea este θ(1) in best case

// Overall complexitatea este O(n)

array[sfarsit]=e

daca sfarsit= cp atunci

sfarsit<-1

altfel

Sfarsit<-sfarsit+1

Sfdaca

sfadauga

Subalgoritm element(array)

//Complexitatea este θ(1)

daca final=0 atunci

{aruncam exceptie}

altfel

return array[inceput]

sfdaca

sfelement

Subalgoritm sterge (array,inceput)

//Complexitatea este θ(1)

daca sfarsit=0 atunci

{aruncam exceptie}

altfel

inceput<-inceput+1

sfdaca

sfsterge

Subalgoritm vida ()

//Complexitatea este θ(1)

daca final=inceput atunci

return Adevarat

altfel

return Fals

sfdaca

sfvida

**Functionalitate lab**

Coada:

Cp-numar intreg dat de la utilizator

Final-numar intreg reprezentand finalul cozii

Aray-Telem\* care reprezinta vectorul dynamic

Inceput-numar intreg dat de la utilizator

Subalgoritm adauga (array,e)

// complexitatea este θ(1)

// preconditii: ca elemental sa fie de TElement

// postconditii:fals-> daca finalul a ajuns la capacitatea impusa nu se va mai adauga in lista ,altfel //adevarat

daca sfarsit+1> cp atunci

afiseaza false si mesajul “Capacitatea este plina”

sfarsit daca

array[final]<-elem

final<-final+1;

afiseaza true

sfadauga