

Linked List Algo

Déclaration :

Structure = Node
head : entier
next : ^Node
Fin structure

Remplir avec n nodes :

fonction Fill(n : entier) : ^Node

Von
i, e : entier
new, mode, prev : ^Node

Début-

Allocate(mode)

mode ← nil

prev ← nil

Pour i de 1 à n faire
lire(e)

Allocate(new)

new[^].head ← e

new[^].next ← nil

if (prev ≠ nil) Alors

prev[^].next ← new

sinon

mode ← new

Fin si

prev ← new

retourner mode

Fin

get number of nodes?

fonction Nnodes (node: ^Node) : entier

Var

count: entier

faux: ^node

Debut

faux ← node

count ← 0

tant que faux ≠ nil faire

count ← count + 1.

faux ← faux.next.

Fin tant que

retourner n

Fin

