

Last name	
First name	
Group	

Grade	/ 5
-------	-----

Algorithmics
Midterm Exam 1 - Part 1
 Undergraduate 1st year S1
 EPITA
 9 Nov. 2020 - 8 : 30

- This is the part 1 of the subject - You have to give back the two parts!
- You must answer on **this subject**.
 - Answer within the provided space. **Answers outside will not be marked.**
 - Pencil answers will not be marked.
- The presentation is marked.

Exercise 1 (Abstract Types: Recursive lists – 5 points)

Consider the algebraic abstract type *recursive list* seen in class and recalled below.

TYPES

list, box

USES

element

OPERATIONS

$emptylist$: $\rightarrow list$
 $head$: $list \rightarrow box$
 $contents$: $box \rightarrow element$
 $first$: $list \rightarrow element$
 $cons$: $element \times list \rightarrow list$
 $tail$: $list \rightarrow list$
 $next$: $box \rightarrow box$

PRECONDITIONS

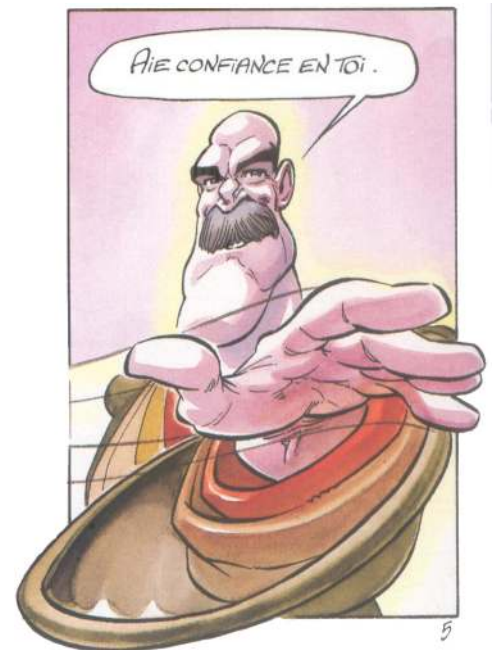
$head(\lambda)$ **is-defined-iaoi** $\lambda \neq emptylist$
 $tail(\lambda)$ **is-defined-iaoi** $\lambda \neq emptylist$
 $first(\lambda)$ **is-defined-iaoi** $\lambda \neq emptylist$

AXIOMS

$first(cons(e, \lambda)) = e$
 $tail(cons(e, \lambda)) = \lambda$
 $contents(head(\lambda)) = first(\lambda)$
 $next(head(\lambda)) = head(tail(\lambda))$

WITH

list λ
 element e



We propose to extend the properties of this type allowing it:

- to search for an element in a list
- to concatenate two lists.

