Contents

1 Module `Liststack` : Last-in-first-out stack data structure. 1

1 Module `Liststack` : Last-in-first-out stack data structure.

This module implements stacks (LIFOs) using OCaml list

`type 'a t`

The type of stacks containing the elements of type `'a`.

`exception Stack_empty`

Raised when `Liststack.pop`[1] or `Liststack.top`[1] is applied to an empty stack.

`val create : unit -> 'a t`

Return a new stack, initially empty.

`val is_empty : 'a t -> bool`

Return `true` if the given stack is empty. Otherwise return `false`.

`val push : 'a -> 'a t -> 'a t`

`push x s` adds the element `x` at the top of stack `s` and returns the new stack.

`val pop : 'a t -> 'a t`

`pop s` removes the top element of stack `s` and return the new stack.

`val top : 'a t -> 'a`

`top s` returns the top element. Stack `s` remains the same as before.

`val print : 'a t -> ('a -> string) -> unit`

Print the stack content to standard stdout.