Contents

1 Module `Listplus` : This module provides some extra useful functions for OCaml list.

`val last : 'a list -> 'a option`

Return the last element of the given list.

`val biggest_prefix : 'a list -> 'a list`

Return the prefix without the last element of the given list. For example, `biggest_prefix [1;2;3] = [1;2]`.

`val truncate : 'a -> 'a list -> 'a list`

`truncate a l` truncates list `l` from the first occurrence of `a`. Return `l` if `a` is not found in `l`.

`val catenate : 'a list -> 'a list -> 'a list`

Catenate two lists.

`val prefixes : 'a list -> 'a list list`

Return a list of all prefixes of the given list in increasing order of length. For example, `prefixes [1;2;3] = [[];[1];[1;2];[1;2;3]]`.

`val suffixes : 'a list -> 'a list list`

Return a list of all suffixes of the given list in decreasing order of length. For example, `suffixes [1;2;3] = [[1;2;3];[2,3];[3],[]]`.

`val print : 'a list -> ('a -> string) -> unit`

Print the list content to standard stdout.

`val print_pre_suffixes : 'a list list -> ('a -> string) -> unit`

Print the prefixes or suffixes of a list to standard stdout.