

Aide-mémoire Caml

Opérateurs booléens : & && or || not

Fonctions sur float : exp log sqrt sin cos tan asin acos atan abs_float

Fonctions de base sur listes : list_length @ hd tl rev map

| Fonctions de conversion | |
|-------------------------|--------------------------|
| string_of_bool | |
| int_of_char | caractère → code "ASCII" |
| char_of_int | code "ASCII" → caractère |
| int_of_float | |
| float_of_int | |
| string_of_int | |
| int_of_string | |
| string_of_float | |
| float_of_string | |

| Lecture et impression | |
|-----------------------|---------------|
| print_char | char → unit |
| print_string | string → unit |
| print_int | int → unit |
| print_float | float → unit |
| print_newline | unit → unit |
| read_line | unit → string |
| read_int | unit → int |
| read_float | unit → float |

| Autres fonctions sur listes | | |
|-----------------------------|------------------------------------|----------------------------|
| do_list | ('a → unit) → 'a list → unit | |
| it_list | ('a → 'b → 'a) → 'a → 'b list → 'a | |
| list_it | ('a → 'b → 'b) → 'a list → 'b → 'b | |
| for_all | ('a → bool) → 'a list → bool | ∀ |
| exists | ('a → bool) → 'a list → bool | ∃ |
| mem | 'a → 'a list → bool | test d'appartenance |
| index | 'a → 'a list → int | indice première occurrence |
| assoc | 'a → ('a * 'b) list → 'b | association |

| Couples | | |
|---------|------------------------------------|------------------------------|
| fst | 'a * 'b → 'a | première composante |
| snd | 'a * 'b → 'b | deuxième composante |
| split | ('a * 'b) list → 'a list * 'b list | découpage liste de couples |
| combine | 'a list * 'b list → ('a * 'b) list | recollement couple de listes |

| Chaines de caractères | | |
|-----------------------|-----------------------------|------------------|
| string_length | string → int | |
| prefix ^ | string → string → string | concaténation |
| sub_string | string → int → int → string | <i>s deb lgr</i> |
| make_string | int → char → string | |

| Manipulation de vecteurs | | |
|--------------------------|-------------------------------|-------------------------|
| vect_length | 'a vect → int | |
| make_vect | int → 'a → 'a vect | |
| make_matrix | int → int → 'a → 'a vect vect | |
| sub_vect | 'a vect → int → int → 'a vect | <i>v deb lgr</i> |
| list_of_vect | 'a vect → 'a list | |
| vect_of_list | 'a list → 'a vect | |
| map_vect | ('a → 'b) → 'a vect → 'b vect | application de fonction |

| Constructions usuelles ; { et } indiquent des éléments optionnels |
|---|
| let {rec} liaison {and liaison} in expression |
| match expression with filtrage |
| try expression with filtrage |
| for nom = expression to expression do expression done |
| if expression then expression {else expression } |

| Exceptions |
|------------------------|
| exception id |
| raise id |
| exception id {of type} |
| raise (id expr) |
| failwith string |

| Motifs de filtrage | |
|---|---|
| id _ motif as id | |
| constante | |
| [] motif :: motif [motif ; motif...motif] | listes |
| motif, ..., motif | couples, n-uples |
| {etiq=motif ; ... ; etiq=motif} | produits ; { et } font partie de la syntaxe |
| motif motif | cas groupés |
| motif when cond | filtrage avec garde |
| cc cnc motif | constructeurs (constant, non constant) |