A TYPE FOR ARRAY INDEXES (40/40 points)

The previous week, we asked you the following question: Consider a non empty array of integers `a`, write a function `min_index : int array -> int` that returns the index of the minimal element of `a`.
As the arrays contain integers and the indices of arrays are also represented by integers, you might have confused an index and the content of a cell. To avoid such a confusion, let us define a type for index (given in the prelude below).
This type has a single constructor waiting for one integer.
For instance, if you want to represent the index 0, use the value `Index 0`.
Defining such a type is interesting because it allows the type-checker to check that an integer is not used where an index is expected (or the converse).

1. Write a function `read : int array -> index -> int` such that `read a (Index k)` returns the k-th element of `a`.
2. Write a function `inside : int array -> index -> bool` such that `inside a idx` is true if and only if `idx` is a valid index for the array `a`.
3. Write a function `next : index -> index` such that `next (Index k)` is equal to `Index (k + 1)`.
4. Consider a non empty array of integers `a`, write a function `min_index : int array -> index` that returns the index of the minimal element of `a`.

THE GIVEN PRELUDE

```
type index = Index of int
```

YOUR OCAML ENVIRONMENT

```
let read a index = match index with
  | Index k -> a.(k)
;;

let inside a index = match index with
  | Index k -> if k >= 0 && k < Array.length a then true else false
;;

let next index = match index with
  | Index k -> Index (k + 1)
;;

let min_index a =
  let rec min_index_rec a idx_compt idx_min=
    if inside a (next idx_compt) = false then idx_min else
    if read a idx_compt < read a idx_min then
      min_index_rec a (next idx_compt) idx_compt else
      min_index_rec a (next idx_compt) idx_min
  in min_index_rec a (Index 0) (Index 0)
;;

```

Evaluate >

Switch >>

Typechecl

Reset Templ

Full-screen |

Check & Sa

Exercise complete (click for details) 40 pts

v Exercise 1: read Completed, 10 pts

```
Found read with compatible type.
Computing read [|12; -7|] (Index 1)
Correct value -7                                                  1 pt
Computing read [|-12; -6; -4; 13; 7; 14; -7; 5; -15; -13; -10|] (Index 10)
Correct value -10                                                 1 pt
Computing read [|0; 4; 6|] (Index 1)
Correct value 4                                                   1 pt
Computing read [|7; 13; 6|] (Index 2)
Computing read [|13; 9; -12; 1; 10|] (Index 2)
Correct value -12                                                      1 pt
Computing read [|-7; 2; 0|] (Index 2)
Correct value 0                                                        1 pt
Computing read [|-9; 6; 12; -13; 11; -14; 8|] (Index 4)
Correct value 11                                                       1 pt
Computing read [|-14; 10; -5; -11; 1; 9; 6; 13; -8; 8|] (Index 6)
Correct value 6                                                        1 pt
Computing read [|-9; 12; 11; 3; 13; 4; 1; -13; 5; -2|] (Index 5)
Correct value 4                                                        1 pt
v Exercise 2: inside                                     Completed, 10 pts
Found inside with compatible type.
Computing inside [|13; 0|] (Index (-2))
Correct value false                                                    1 pt
Computing inside [|-14; -6; 7; 14|] (Index 6)
Correct value false                                                    1 pt
Computing inside [|6; 12; -12; 14; -10; -9|] (Index 1)
Correct value true                                                     1 pt
Computing inside [|-1; -6; 8; 6; -9; -2; 11|] (Index 8)
Correct value false                                                    1 pt
Computing inside [|1; -10; 2; -13|] (Index (-3))
Correct value false                                                    1 pt
Computing inside [|-6; 10; 9; -9|] (Index 3)
Correct value true                                                     1 pt
Computing inside [|7; 13; -12; 3; 10; 11; -4|] (Index 5)
Correct value true                                                     1 pt
Computing inside [|11; -9; -14; 9|] (Index 0)
Correct value true                                                     1 pt
Computing inside [|-10; 3; 13; 5; 4; -1; -7; -2; 10|] (Index 12)
Correct value false                                                    1 pt
Computing inside [|-12; 10; -4; -10; 3; -8; -9; -6; -13; 4; 1|] (Index (-5))
Correct value false                                                    1 pt
v Exercise 3: next                                       Completed, 10 pts
Found next with compatible type.
Computing next (Index 8)
Correct value (Index 9)                                                1 pt
Computing next (Index (-14))
Correct value (Index (-13))                                            1 pt
Computing next (Index (-49))
Correct value (Index (-48))                                            1 pt
Computing next (Index (-32))
Correct value (Index (-31))                                            1 pt
Computing next (Index 30)
Correct value (Index 31)                                               1 pt
Computing next (Index 33)
Correct value (Index 34)                                               1 pt
Computing next (Index (-3))
Correct value (Index (-2))                                             1 pt
Computing next (Index (-21))
Correct value (Index (-20))                                            1 pt
Computing next (Index (-30))
Correct value (Index (-29))                                            1 pt
Computing next (Index 33)
Correct value (Index 34)                                               1 pt
v Exercise 4: min_index                                  Completed, 10 pts
Found min_index with compatible type.
Computing min_index [|-2; -12; -6; -11; -3; 5; 2; -1; 14; -9; 3|]
Correct value (Index 1)                                                1 pt
Computing min_index [|14; -6; 10|]
Correct value (Index 1)                                                1 pt
Computing min_index [|4; 5; -5; -14; 0; -2|]
Correct value (Index 3)                                                1 pt
```

Rechercher un cours

```
Computing min_index [|-5; -11; 0; 4; 9; -6; 5; -4; -5; 7; -10; 12|]
Correct value (Index 1)                                                1 pt
Computing min_index [|-11; 3; 2|]
Correct value (Index 0)                                                1 pt
Computing min_index [|-4; -7; 3; 2; -14; 8; -6|]
Correct value (Index 4)                                                1 pt
Computing min_index [|-4; 5; 11|]
Correct value (Index 0)                                                1 pt
Computing min_index [|-15; -9; 8; 4; -13|]
Correct value (Index 0)                                                1 pt
Computing min_index [|9; -13; -4; 3; -7; -10; -1|]
Correct value (Index 1)                                                1 pt
```