```
theory Diff_Arr_Length
  imports Diff_Arr_Rel Master_Assn
begin

(* TODO: Use Locale instead of context *)
context
begin

type_synonym 'a diff_arr = "'a cell ref"

qualified partial_function (heap) length ::
  "('a::heap) diff_arr ⇒ nat Heap"
where
  "length diff_arr = do {
      cell ← !diff_arr;
      case cell of
        Array array   ⇒ Array.len array
      | Upd _ _ diff_arr ⇒ length diff_arr
  }"

lemma length [sep_heap_rules]:
  "<master_assn t * ↑(t ⊢ xs ~ diff_arr)>
     length diff_arr
   <λlen. master_assn t * ↑(len = List.length xs)>"
  sorry

end

end
```