285   (Dutch national flag) Given a variable

$$flag: [*(red, white, blue)]$$

sort it so that all *red* values are first, all *white* values are in the middle, and all *blue* values are last. The only way allowed to change *flag* is to use

$$swap = \langle i, j: \Box flag \cdot flag := i \rightarrow flag\ j \mid j \rightarrow flag\ i \mid flag \rangle$$

After trying the question, scroll down to the solution.

§   One way to say the specification is

  flagsort

= *perm flag flag'*

  ∧ ∃a, b· $0{\le}a{\le}b{\le}\#flag$ ∧ *flag'*$(0,..a)$: *red* ∧ *flag'*$(a,..b)$: *white* ∧ *flag'*$(b,..\#flag)$: *blue*

where

   perm = ⟨L, M· ∀x· ¢($§i$: □L· $L\,i = x$) = ¢($§i$: □M· $M\,i = x$)⟩

We introduce 3 index variables i , j , and k so that in the middle of execution, the picture looks like this (using R for *red* , W for *white* , B for *blue* , and X for unknown):

R R R W W W X X X B B B

  ↑ ↑ ↑

  i j k

We need to sort the segment from i to k , given that from i to j is all *white* .

  flagsortsegment

= $0{\le}i{\le}j{\le}k{\le}\#flag$ ∧ *flag*$(i,..j)$: *white*

⇒ *perm flag flag'*

  ∧ ∃a, b· $i{\le}a{\le}b{\le}k$ ∧ *flag'*$(i,..a)$: *red* ∧ *flag'*$(a,..b)$: *white* ∧ *flag'*$(b,..k)$: *blue*

We can now refine:

flagsort ⟸ $i:= 0$. $j:= 0$. $k:= \#flag$. *flagsortsegment*

flagsortsegment ⟸ **if** $j{=}k$ **then** *ok*

     else if *flag*$(k{-}1){=}blue$ **then** $k:= k{-}1$. *flagsortsegment*

      else if *flag j* = *white* **then** $j:= j{+}1$. *flagsortsegment*

       else if *flag j* = *red*

        then *swap i j*. $i:= i{+}1$. $j:= j{+}1$. *flagsortsegment*

        else $k:= k{-}1$. *swap j k*. *flagsortsegment* **fi fi fi fi**

The line that tests *flag*$(k{-}1){=}blue$ is unnecessary; maybe it improves the average execution time.