The Dutch National Flag Problem

Roland Backhouse

April 23, 2001

Outline

An algorithm is developed for solving the Dutch National Flag Problem.

The development illustrates the use of invariant properties in the design of loops.

The solution to the Dutch National Flag Problem will be used later in the development of an algorithm for finding the best k values in an unordered set.

Problem Statement

Informal

The problem concerns the control of a robot that has the task of sorting a number of coloured pebbles contained in a row of buckets.

The buckets are arranged in front of the robot and each contains exactly one pebble coloured either red, white or blue.

The robot is equipped with two arms, on the end of each of which is an eye. Using its eyes the robot can determine the colour of the pebble in each bucket; it can also swap the pebbles in any pair of buckets.

The problem is to issue a sequence of instructions to the robot causing it to rearrange the pebbles into the order of the colours in the Dutch National Flag, namely red, white and blue.

Colour Determination

We assume a number of values are stored, these values being indexed by numbers i such that $M \le i < N$.

We assume that $M \le N$ but do not assume that $M < N$; if $M = N$ then the number of stored values is zero.

We assume that boolean-valued functions red, $white$ and $blue$ on the indices determine the colour of the stored values. That is, $red.i$ means the value indexed by i is red, and similarly for $white.i$ and $blue.i$.

We do *not* assume that there is at least one value of each colour.

Swapping Values

Swapping the pebbles in buckets i and j is effected by executing $swap(i,j)$.

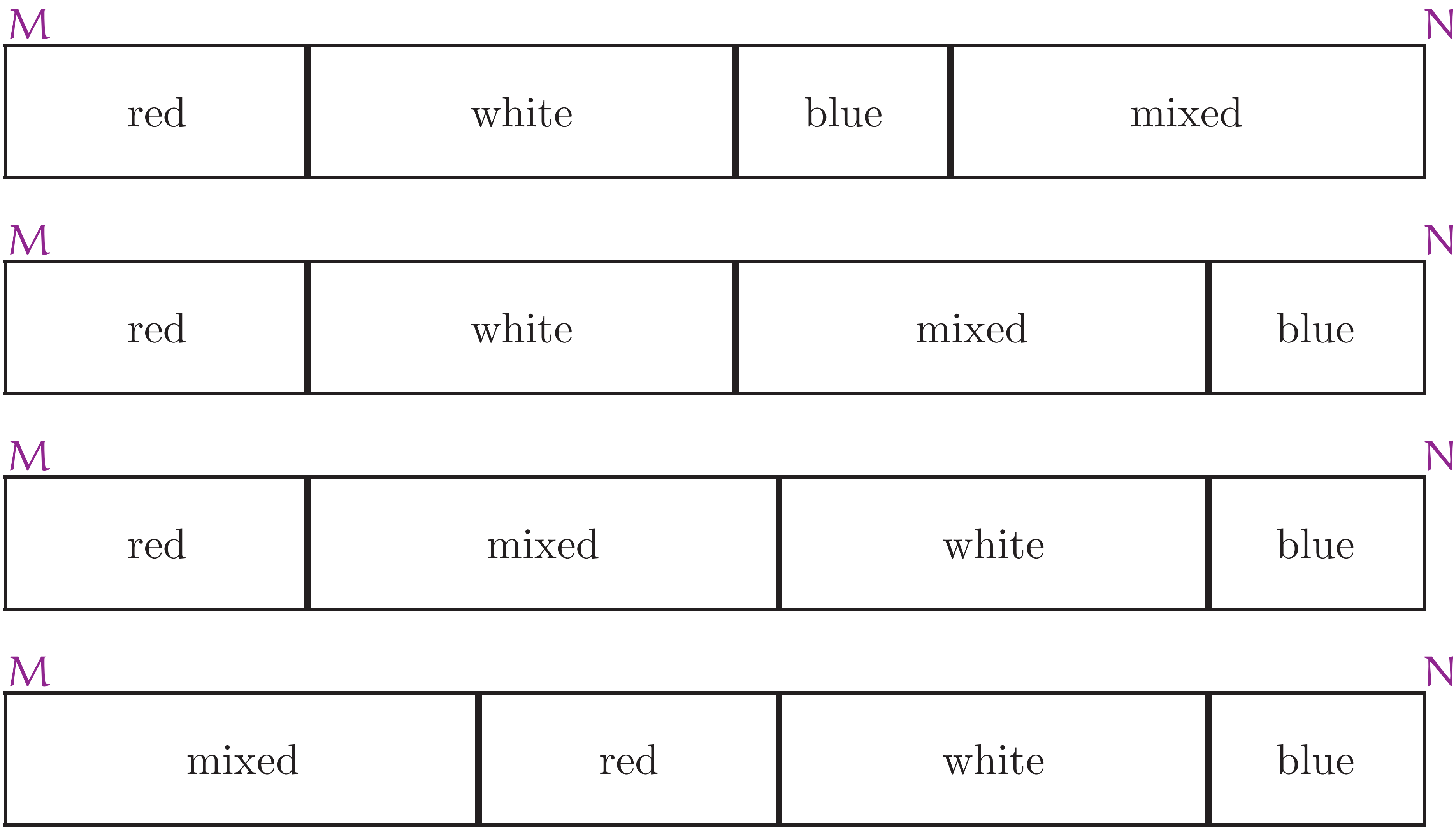
It is convenient to assume that $swap(i,i)$ is valid and has no effect on the state of the stored values (as predicted by the formal specification of $swap$).

Postcondition

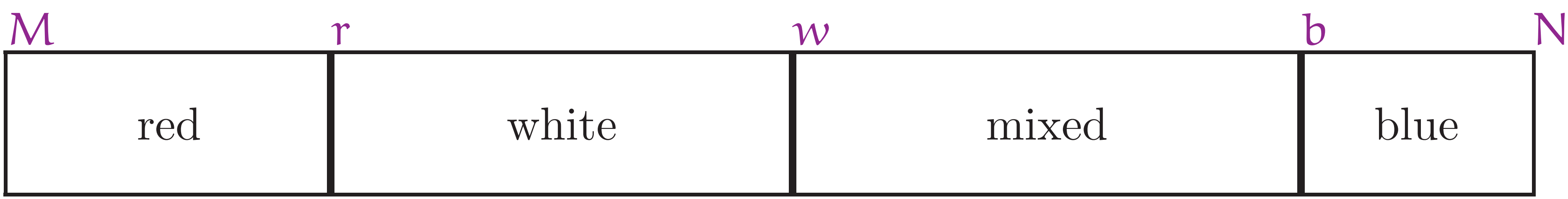
We are required to construct a program, making use exclusively of the above operations together with simple arithmetic operations on indices, that will rearrange the stored values in such a way that on termination there are indices r and w such that

$$\begin{aligned} & M \leq r \leq w \leq N \\ \wedge\; & \langle \forall i \mid M \leq i < r : \mathrm{red}.i \rangle \\ \wedge\; & \langle \forall i \mid r \leq i < w : \mathrm{white}.i \rangle \\ \wedge\; & \langle \forall i \mid w \leq i < N : \mathrm{blue}.i \rangle \quad . \end{aligned}$$

Possible Choices of Invariant



Chosen Invariant



Goal: design a simple loop (together with its initialisation) that maintains invariant the property

$$
\begin{aligned}
& M \leq r \leq w \leq b \leq N \\
\wedge\ & \langle \forall i \mid M \leq i < r : \mathrm{red}.i \rangle \\
\wedge\ & \langle \forall i \mid r \leq i < w : \mathrm{white}.i \rangle \\
\wedge\ & \langle \forall i \mid b \leq i < N : \mathrm{blue}.i \rangle
\end{aligned}
$$

whilst decreasing the size $b - w$ of the "mixed" segment.

Skeleton algorithm

$\{\ M \le N\ \}$

$r,w,b := M,M,N$

$\{$ **Invariant:** $M \le r \le w \le b \le N$

$\wedge \langle \forall i \mid M \le i < r : red.i \rangle$

$\wedge \langle \forall i \mid r \le i < w : white.i \rangle$

$\wedge \langle \forall i \mid b \le i < N : blue.i \rangle$

Bound function: $b - w$ $\}$;

do $w < b \rightarrow$ reduce $b - w$ whilst maintaining the invariant

od

$\{\quad M \le r \le w \le N$

$\wedge \langle \forall i \mid M \le i < r : red.i \rangle$

$\wedge \langle \forall i \mid r \le i < w : white.i \rangle$

$\wedge \langle \forall i \mid w \le i < N : blue.i \rangle\ \}$

Making progress

if $white.w \rightarrow w := w{+}1$

$\Box$ $red.w \rightarrow swap(r,w)\ ;\ r,w := r{+}1\,, w{+}1$

$\Box$ $blue.w \rightarrow swap(b{-}1\,, w)\ ;\ b := b{-}1$

fi

The Algorithm

$\{\ M \leq N\ \}$

$r,w,b := M,M,N$

$\{$ **Invariant:** $M \leq r \leq w \leq b \leq N \wedge \langle \forall i \mid M \leq i < r : red.i \rangle$

$\wedge \langle \forall i \mid r \leq i < w : white.i \rangle \wedge \langle \forall i \mid b \leq i < N : blue.i \rangle$

Bound function: $b-w$ $\}$;

do $w < b \rightarrow$ if $white.w \rightarrow w := w+1$

$\square$ $red.w \rightarrow swap(r,w)\ ;\ r,w := r+1, w+1$

$\square$ $blue.w \rightarrow swap(b-1, w)\ ;\ b := b-1$

fi

od

$\{\quad M \leq r \leq w \leq N \wedge \langle \forall i \mid M \leq i < r : red.i \rangle$

$\wedge \langle \forall i \mid r \leq i < w : white.i \rangle \wedge \langle \forall i \mid w \leq i < N : blue.i \rangle\ \}$