1 Initialization Blocks Handler

1.1 Handler Algorithm

Algorithm 1: Handle

Input: L, B, R, M

$A_L \leftarrow \{l \in L \mid (\neg\exists b \in B)(l.id = b.id)\}$;
$A_R \leftarrow \{r \in R \mid (\neg\exists b \in B)(r.id = b.id)\}$;
$D_L \leftarrow \{b \in B \mid (\neg\exists l \in L)(b.id = l.id)\}$;
$D_R \leftarrow \{b \in B \mid (\neg\exists r \in R)(b.id = r.id)\}$;
$D \leftarrow D_L \cap D_R$;
$IB_L \leftarrow \{n \in A_L \mid n.type = INITBLOCK\}$;
$IB_R \leftarrow \{n \in A_R \mid n.type = INITBLOCK\}$;
$IB_B \leftarrow \{n \in D \mid n.type = INITBLOCK\}$;
$matches \leftarrow \emptyset$;
if $|IB_L| = 1 \wedge |IB_R| == 1 \wedge |IB_B| = 1$ **then**
  $matches \leftarrow matches \cup ()$;
else
  foreach $b \in IB_B$ **do**
    $l \leftarrow findFirst(l \in IB_L \rightarrow l.body \approx b.body)$;
    $r \leftarrow findFirst(r \in IB_R \rightarrow r.body \approx b.body)$;
    $IB_L \leftarrow IB_L - l$;
    $IB_R \leftarrow IB_R - r$;
    if $l \neq null \wedge r \neq null$ **then**
      $matches \leftarrow matches \cup (l, b, r)$;
    end
  end
  foreach $l \in IB_L$ **do**
    $r \leftarrow findFirst(r \in IB_R \rightarrow r.body \approx l.body)$;
    $IB_R \leftarrow IB_R - r$;
    if $r \neq null$ **then**
      $matches \leftarrow matches \cup (l, null, r)$;
    end
  end
end
foreach $(l, b, r) \in matches$ **do**
  $m \leftarrow find(m \in M \rightarrow m.body = l.body)$;
  $m.body \leftarrow$ `textualMerge(`$l.body$, $b.body$, $r.body$`)`;
  $m \leftarrow find(m \in M \rightarrow m.body = r.body)$;
  `removeNode(`m, M`)`;
end