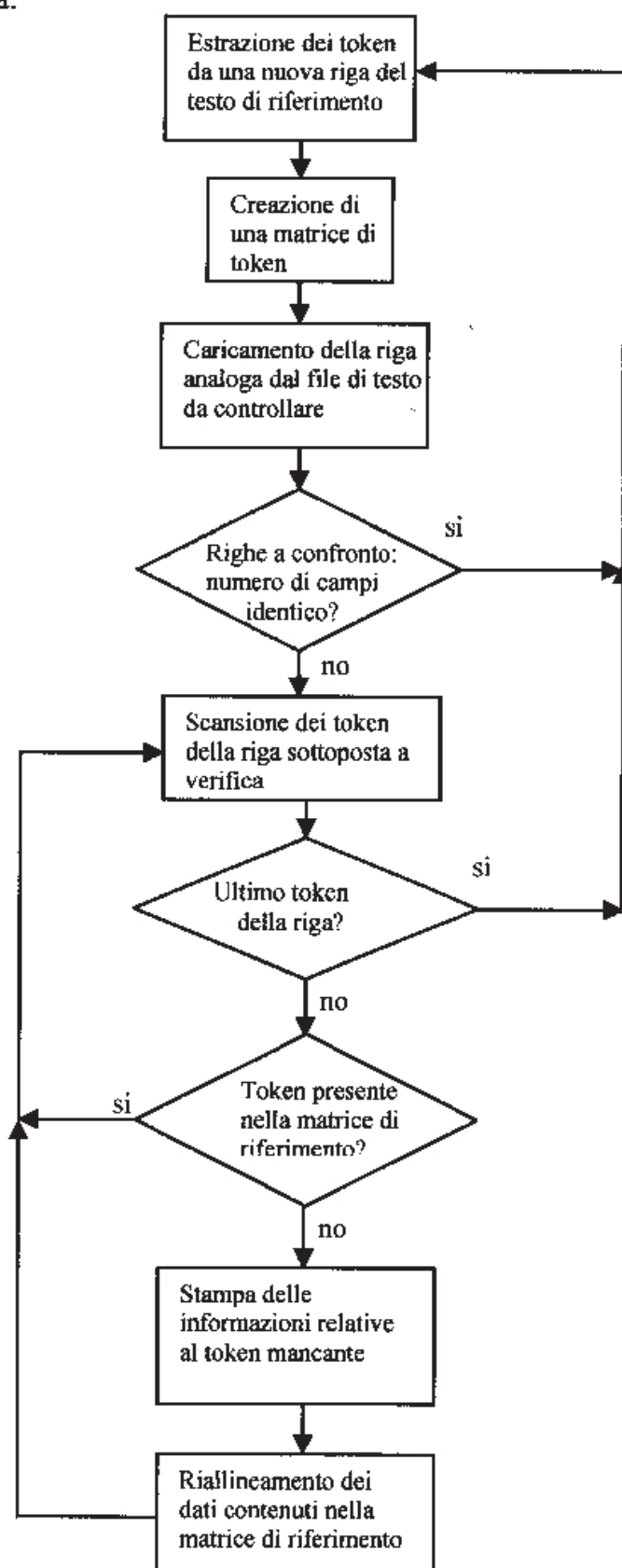



```
tf++
if ($tf ~ ^#/ || $tf ~ ^@/ || $tf ~ ^%/ || $tf ~ ^$/ || $tf ~ ^E/)
  continue
sp = split($tf, cp, "_")
part = cp[1]
if (part == tab[tf])
  continue
else
  {
  print "token *" tab[tf] "* mancante alla riga "
    count " in posizione " tf
  rl = tf
  while (rl <= campi)
    {
    fw = rl
    fw++
    tab[rl] = tab[fw]
    rl++
    }
  tf--
  continue
  }
}
```

Tav. 71: Listato di *checkLine*, script in GAWK.

Il principio di funzionamento di questo script riprende a grandi linee quello del programma precedente, arricchendolo tuttavia di nuove funzionalità. Anch'esso, infatti, opera mediante una funzione di confronto tra due testi, di cui uno utilizzato come riferimento, non limitandosi però al semplice conteggio delle componenti lessicali. Il programma, infatti, permette di individuare esattamente la posizione di una lacuna all'interno di un testo, sfruttando un confronto sistematico dei dati presenti in ogni singola riga. Ciò è possibile mediante l'utilizzo di una matrice composta dai token estratti da ogni singola riga del testo di riferimento e indicizzata dal puntatore al token stesso. Scendendo nel dettaglio, il programma si articola in tre passaggi: il primo, di preparazione, prevede la lettura dal file di riferimento di una intera riga di testo, l'estrazione dei token in essa presenti e la loro inserzione in una matrice indicizzata. Il secondo passaggio, anch'esso preparatorio, prevede invece l'acquisizione dell'identica riga di testo dal file oggetto di verifica, sfruttando il comando *getline*. A ciò fa seguito un rapido controllo sulla quantità dei token presenti nelle due righe caricate in memoria: valori uguali rappresentano un chiaro indice di assenza di problemi, con conseguente possibilità di passaggio immediato alla riga di *input* successiva. In caso contrario, il sistema prevede l'attivazione dell'algoritmo preposto all'individuazione della lacuna. Questo algoritmo, mediante un confronto tra ogni singolo token presente nella riga del testo da verificare ed il contenuto della matrice di riferimento, consente di definire con estrema precisione il numero della linea che ospita la lacuna, il token mancante e la sua posizione all'interno della linea stessa. Naturalmente, poiché la lacuna è fonte di divergenza tra le due entità testuali utilizzate dal programma, si rivela necessario operare un riallineamento della struttura della matrice sulla base delle lacune indi-

viduate, al fine di garantire al sistema il proseguimento regolare dell'attività di controllo fino al termine della riga.



Tav. 72: Diagramma di flusso di checkLine, script in GAWK.

7.5.1.6 IL "TRANSORD". [MT] Come accennato a inizio paragrafo, oltre agli importantissimi sistemi di controllo, nel corso dell'opera si è rivelato necessario possedere un livello