

SPIEGAZIONE DEL PROGRAMMA IN SHELL BASH

Il programma può essere suddiviso nelle tre macro aree sequenziali: analisi capillare della riga di comando, stampa d'informazioni di carattere generale ed infine esecuzione della stringa di comando; descriviamo nel dettaglio le varie fasi.

Analisi capillare

La prima parte del codice prevede l'inizializzazione di tutte le variabili considerate nel programma, in particolare esse sono raggruppare in base alla tipologia ed alla loro "affinità" nelle varie operazioni (variabili usate nelle stesse operazioni sono raggruppate insieme). Si passa poi alla "cattura" della riga di comando ed alla successiva analisi su base di singola parola: l'idea che è quella di riuscire ad interpretare la parola sotto osservazione, distinguendola come parola "speciale" (redirezione), oppure come una parola "comune" (un comando bash, un file, ...); per ottenere questo risultato si passa alla fase principale di questa macro aera, l'analisi capillare carattere per carattere.

Questa analisi prevede di eseguire una serie di "domande", atte a determinare se il carattere sia: in prima posizione della riga, in prima posizione della parola, se è un carattere speciale e quali siano i caratteri (o la parola) che precedono il carattere corrente; in base alle risposte che si ottengono il programma assume comportamenti differenti che lo portano ad eseguire specifiche routine.

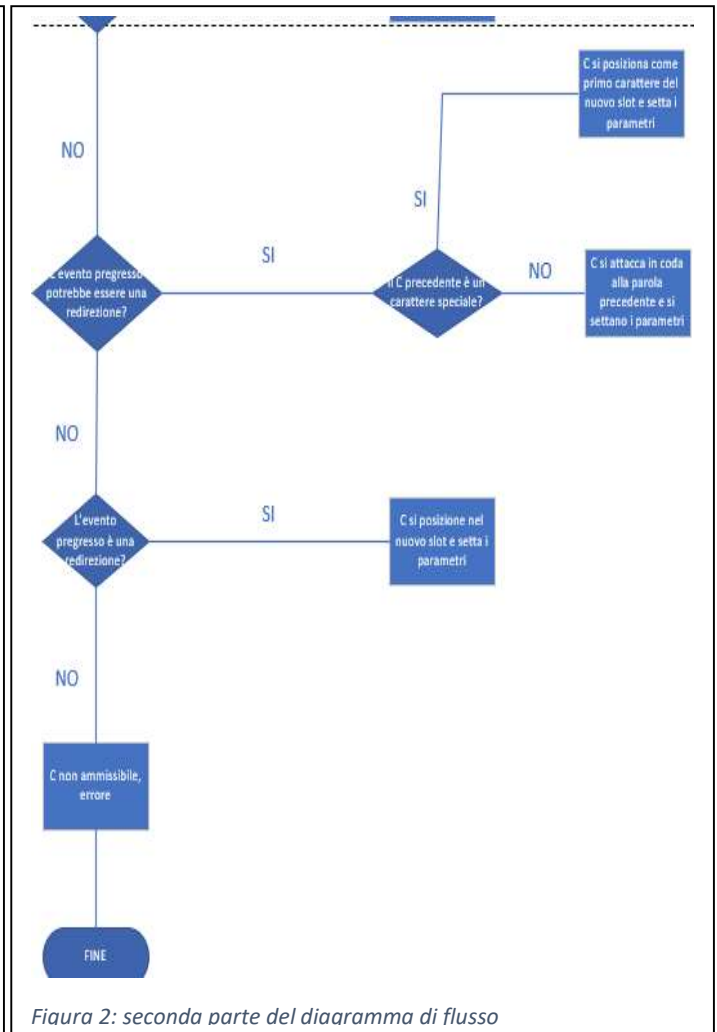
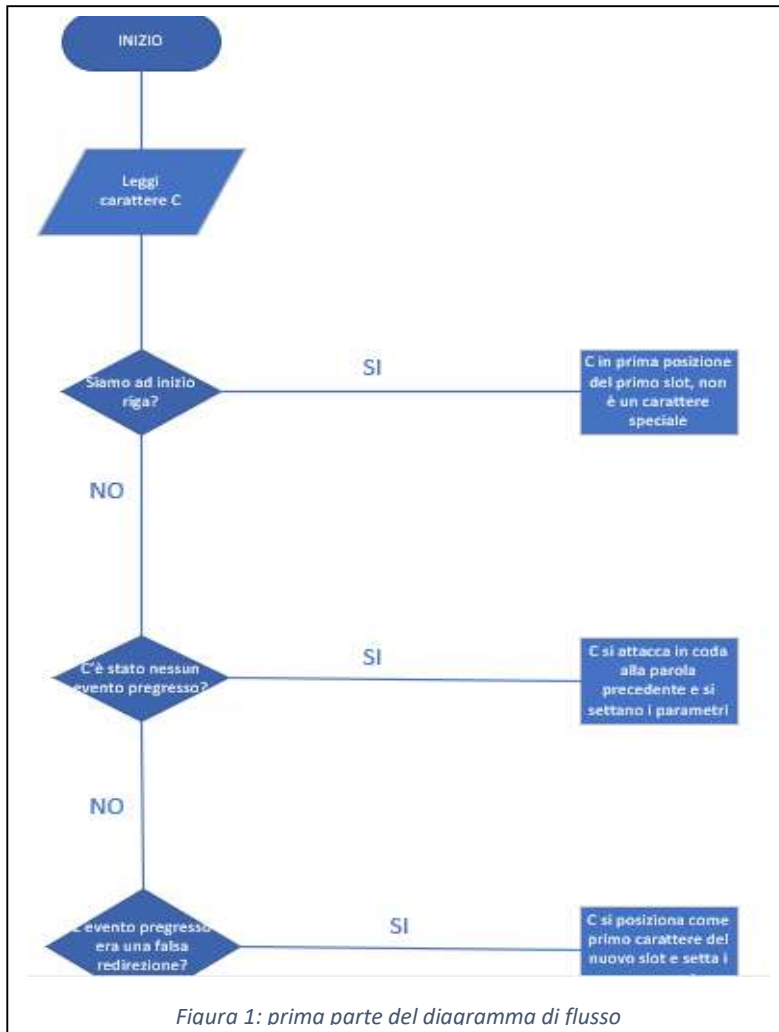
Per comprendere al meglio la natura dell'analisi capillare forniamo di seguito uno schema sul riconoscimento delle parole speciali.

Valore parametri passati	analisi carattere					Valore parametri presenti
$A = R = M = 0$	1	>	<	2	&	$R = 0, A = M = 1$
	↓	↓	↓	↓	↓	
$R = 0, A = M = 1$	>, <	>, &, \\\	<, &, \\\	>, <	>, <	$M = 2, A = R = 1$
	↓	↓	↓	↓	↓	
$M = 2, A = R = 1$	\\	\\	\\	&	\\	$M = 3, A = R = 1$
				↓		
$M = 3, A = R = 1$				1		$M = 4, A = R = 1$
				↓		
$M = 4, A = R = 1$				\\		$M = 5, A = R = 1$

Dove:

- A = allerta, il programma è messo in allerta in quanto il carattere sotto osservazione è l'inizio di una parola speciale;
- M = memoria, tiene traccia delle analisi effettuate;
- R = redirection, certifica se la parola è un comando di redirezione
- \\\ = la parola è stata riconosciuta come parola di codice, una redirezione

La sequenza mostrata in precedenza si verifica nel caso ideale, ovvero quando la parola passata è un comando di redirezione, ma nel caso in cui non lo sia si assume un determinato comportamento, in base ai valori attuali dei parametri; di seguito presentiamo, a titolo di esempio, un diagramma di flusso per le decisioni assunte sul solo carattere "1".



Informazioni generali

Durante la fase precedente si andranno a delineare una serie di utili informazioni per facilitare la comprensione del tipo di comando che è stato inserito, in particolare si avranno informazioni su: numero di caratteri speciali per tipologia, dimensione dell'array di parsing, mappatura delle parole, la riga di comando ricevuta in input ed il parsing eseguito su di essa.

Osserviamo che la mappatura svolge anche un ruolo attivo in quanto esso è un'array di 14 posizioni in cui, per ogni posizione, corrisponde un determinato comando di reindirizzamento, come illustrato di seguito.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
\		>	<	>>	<<	> &	< &	1 <	1 >	& <	& >	2 & < 1	2 & > 1

Esecuzione della Shell Bash

Grazie all'ausilio della mappatura è sufficiente usare l'operatore *case* che seleziona la tipologia di esecuzione.

Osservazione

Sotto alcuni aspetti questa Shell è più avanzata di una Shell normale, in quanto essa è in grado di capire se sono presenti errori sintattici nella riga di comando ricevuta in ingresso, prima ancora di tentare di eseguirla: corregge la stringa se l'errore è dovuto ha caratteri di spaziatrice, altrimenti segnala errore.