

# Documentazione Macchina Di Turing Universale

Vincenzo Susso

24 maggio 2021

# Indice

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduzione</b>  | <b>2</b>  |
| <b>2</b> | <b>Requisiti e Descrizione delle Cartelle del Progetto</b> | <b>4</b>  |
| <b>3</b> | <b>Programma - Codificatore MdT</b>                        | <b>5</b>  |
| 3.1      | Descrizione GUI Codificatore MdT . . . . .                 | 5         |
| 3.2      | Creazione della Stringa che Indica una MdT . . . . .       | 7         |
| 3.2.1    | Esempio di Stringa . . . . .                               | 8         |
| <b>4</b> | <b>Macchina Di Turing Universale</b>                       | <b>9</b>  |
| 4.1      | Come Inserire L'Input . . . . .                            | 9         |
| 4.2      | Spiegazione Macchina Di Turing Universale . . . . .        | 10        |
| <b>5</b> | <b>Esempi Simulazione</b>                                  | <b>12</b> |
| 5.1      | Esempio 1 . . . . .  | 12        |
| 5.2      | Esempio 2 . . . . .  | 14        |

# Capitolo 1

## Introduzione

Una **macchina di Turing** (MdT) è un modello computazionale che è stato introdotto nel 1936 dal matematico *Alan Turing* per poter rispondere al problema di decidibilità proposto dal matematico *David Hilbert*.

Una macchina di Turing è formata da un nastro potenzialmente infinito in cui è presente una testina; la testina può effettuare le seguenti azioni:

- Leggere un carattere sul nastro;
- Scrivere un carattere sul nastro;
- Spostarsi sul nastro a destra o a sinistra.

Formalmente una macchina di Turing  $M$  è identificata da una 7-upla:

$$M = (Q, \Sigma, \Gamma, \delta, q_0, \Diamond, F)$$

ove:

- $Q$  è un insieme finito e non vuoto di elementi chiamati **stati**, e proprio per questo,  $Q$  è chiamato **insieme degli stati**;
- $\Sigma$  è un insieme finito e non vuoto di elementi chiamati **simboli**; l'insieme  $\Sigma$  è chiamato **alfabeto di input**;
- $\Gamma$  è chiamato **alfabeto del nastro**;
- $\delta$  è la **funzione di transizione** che serve a descrivere la macchina di Turing ed è specificata nel seguente modo:

$$\delta : Q \times (\Sigma \cup \Gamma) \rightarrow Q \times (\Sigma \cup \Gamma) \times \{L, R\}$$

- $q_0 \in Q$  rappresenta lo **stato iniziale** della macchina di Turing  $M$ ;
- $\diamond \in \Gamma$  è un carattere utilizzato per indicare che una determinata cella del nastro di  $M$  non contiene nulla;
- $F \subseteq Q$  è un insieme che contiene tutti gli **stati di accettazione** di  $M$ .

Le macchine di Turing sono hard-wired, ciò significa che ogni macchina di Turing esegue un solo programma, questo rappresenta una limitazione in quanto i computer reali sono riprogrammabili.

Una **macchina di Turing universale** simula ogni altra macchina di Turing  $M$ ; generalmente una macchina di Turing universale ha tre nastri: il primo è usato per contenere la descrizione della macchina di Turing  $M$  che si vuole simulare; il secondo è usato per contenere l'input della macchina di Turing  $M$ ; infine, il terzo nastro è usato per la computazione vera e propria.

## Capitolo 2

# Requisiti e Descrizione delle Cartelle del Progetto

Questo progetto è diviso in due parti. La prima parte, infatti, è costituita da un programma scritto in *Python* che prende in input una macchina di Turing  $M'$  e restituisce una stringa che identifica  $M'$ ; la seconda parte del progetto è costituita dalla Macchina di Turing Universale  $M$  costruita in *JFLAP*; i requisiti, pertanto, per poter eseguire l'intero progetto sono i seguenti:

- **Python 3.9.1** o superiore (per poter eseguire il programma);
- **tkinter 8.6** o superiore (è una libreria Python utilizzata per implementare l'interfaccia grafica del programma);
- **JFLAP 8.0 Beta** (per poter eseguire la macchina di Turing universale  $M$ ).

Nella cartella del progetto esistono diversi file e cartelle, ovvero:

- Il file *Documentazione.pdf* contiene la documentazione dell'intero progetto;
- La cartella *Latex* contiene i file utilizzati per generare questa documentazione, all'interno della cartella *Latex* è presente la cartella *Images* che contiene le immagini che sono presenti in questa documentazione;
- Il file *main.py* è il programma che è necessario eseguire per poter ottenere la stringa che dovrà essere messa nella macchina di Turing universale  $M$  per poter simulare la macchina di Turing  $M'$ ;
- La cartella *Resources* contiene alcuni file che sono utilizzati dal programma specificato in precedenza;
- Il file *UniversalTuringMachine.jflap* è il file che è necessario aprire con *JFLAP 8.0 Beta* per poter utilizzare la macchina di Turing Universale  $M$ .

# Capitolo 3

## Programma - Codificatore MdT

### 3.1 Descrizione GUI Codificatore MdT

Come detto in precedenza il programma è stato scritto in *Python* e dispone di un'interfaccia grafica (GUI) in modo da facilitare l'interazione con l'utente. Appena avviato il programma la finestra che appare è la seguente:

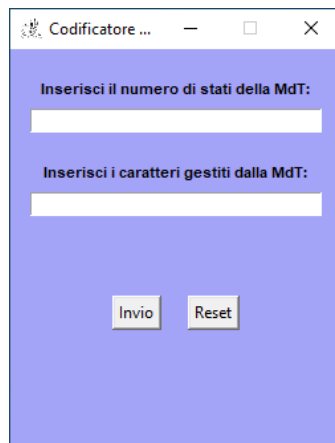


Figura 3.1: Schermata Iniziale

Nella figura 3.1 sono presenti due form in cui inserire l'input; in particolar modo, nel primo form è necessario inserire il numero di stati della macchina di Turing che si vuole simulare (è necessario inserire un numero intero maggiore di 0); nel secondo form devono essere inseriti i caratteri che vengono letti dalla macchina di Turing che si vuole simulare, i caratteri devono appartenere all'alfabeto internazionale, devono essere separati dal carattere "-", devono essere minuscoli e non ci devono essere errori. Se almeno uno di questi vincoli viene violato, apparirà la seguente schermata di errore:

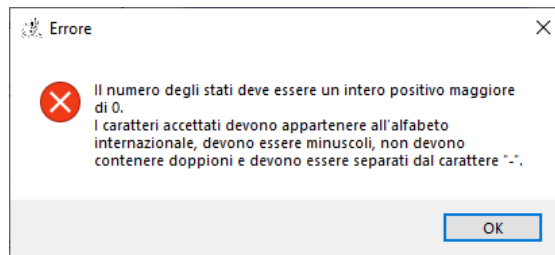


Figura 3.2: Errore Schermata Iniziale

Nella schermata presente nella figura 3.1 sono presenti oltre i due form anche due bottoni; il bottone reset permette di eliminare il testo inserito in precedenza nei due form; una volta riempiti i form correttamente, il bottone *invio* permette di passare alla seguente schermata:

| Stato Finale | Stati | b | a | c |
|--------------|-------|---|---|---|
|              | q0    |   |   |   |
|              | q1    |   |   |   |
|              | q2    |   |   |   |

Figura 3.3: Schermata intermedia

Nella schermata presente nella figura 3.3 è presente una tabella di transizione in cui è possibile descrivere la macchina di Turing  $M$ . Inoltre, lo stato  $q_0$  è sempre stato iniziale. La prima colonna della tabella è utilizzata per indicare gli stati finali della macchina di Turing; se un determinato stato, è uno stato finale allora utilizzeremo un  $1$  altrimenti useremo uno  $0$ ; per esempio, per indicare che lo stato  $q_1$  è stato finale nella cella alla sua sinistra inseriremo un  $1$ .

Nella tabella vengono indicati i caratteri in cima e gli stati al lato; è necessario, pertanto, indicare per ogni cella in cui si incrocia uno stato e un carattere la relativa transizione che avviene leggendo quel determinato carattere in quel determinato stato. Per indicare la transizione è necessario indicare prima lo stato seguito dal carattere che viene scritto e il movimento della testina (è necessario inserire " $l$ " per indicare che la testina si sposta di una cella a sinistra o " $r$ " per indicare che la testina si sposta a destra); tutti questi elementi devono essere separati dal simbolo "-". Per indicare che non avviene nessuna transizione è necessario inserire il simbolo "/".

Dopo aver riempito tutte le caselle della tabella, è possibile premere il tasto "*invio*", se qualche errore è stato commesso, apparirà la seguente schermata di errore:

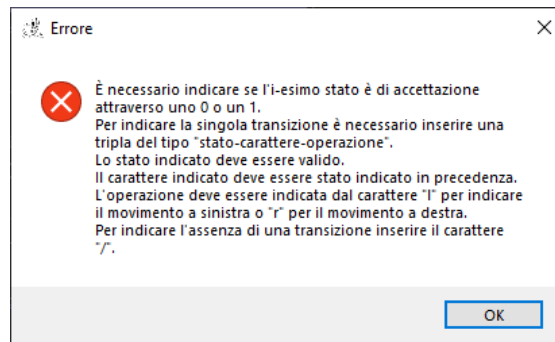


Figura 3.4: Errore Schermata intermedia

Se invece, tutto è stato inserito correttamente, dopo la schermata di invio apparirà la seguente finestra:

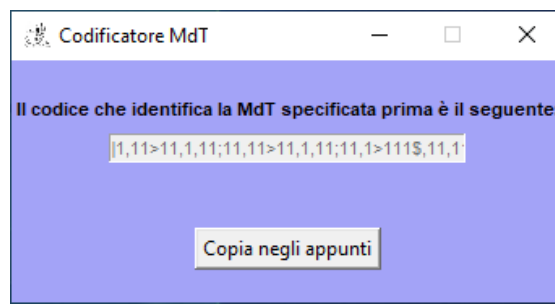


Figura 3.5: Schermata finale

Una volta arrivati alla finestra rappresentata in figura 3.5 è necessario cliccare sul pulsante "*copia negli appunti*" in modo da copiare la stringa che identifica la macchina di Turing inserita; questa stringa sarà necessaria quando utilizzeremo la macchina di Turing universale.

## 3.2 Creazione della Stringa che Indica una MdT

La stringa che indica una macchina di Turing  $M$  contiene diversi simboli che indicano le transizioni eseguite da  $M$ .

Gli stati della macchina di Turing  $M$  sono codificati in alfabeto unario utilizzando il simbolo "1", per esempio: indicheremo con la stringa "1" lo stato  $q_0$ , con la stringa "11" lo stato  $q_1$ , con la stringa "111" lo stato  $q_2$  e così via; il discorso è analogo per l'alfabeto di input della macchina di Turing  $M$ , infatti, anche questo è codificato in alfabeto unario utilizzando il simbolo 1, per esempio: indicheremo con la stringa "1" il carattere  $a$ , con la stringa "11" il carattere  $b$ , con la stringa "111" il carattere



$c$  e così via. Per le codificare le operazioni fatta dalla testina a fine computazione, viene utilizzata la stringa "1" per indicare il movimento della testina a sinistra e la stringa "11" per indicare il movimento della testina a destra. Altri caratteri che vengono usati sono:

- Il carattere "|" viene usato per indicare l'inizio delle transizioni (è sempre il primo carattere della stringa);
- Il carattere "," viene usato per separare i singoli elementi della transizione;
- Il carattere ">" viene usato per separare la prima parte della computazione (quando si legge un carattere dal nastro in un certo stato) con la seconda parte della computazione (in cui un carattere viene scritto sul nastro, avviene un movimento della testina e avviene il cambio di stato);
- Il carattere ";" viene usato per separare le transizioni;
- Il carattere "\$" viene usato per indicare che un determinato stato è anche stato finale.

### 3.2.1 Esempio di Stringa

Prendiamo per esempio questa stringa che descrive una macchina di Turing:

|1, 1 > 11, 1, 11; 11, 11 > 111\$, 11, 11; 11, 1 > 11, 1, 11;

Adesso analizzeremo le singole transizioni che sono separate dal simbolo ";":

1. La transizione 1, 1 > 11, 1, 11 indica che partendo dallo stato  $q_0$  e leggendo il carattere  $a$ , la computazione prevede che si scriva il carattere  $a$ , ci si sposti con la testina a destra e si arrivi allo stato  $q_1$ ;
2. La transizione 11, 11 > 111\$, 11, 11 indica che partendo dallo stato  $q_1$  e leggendo il carattere  $b$ , la computazione preveda che si scriva il carattere  $b$ , ci si sposti a destra con la testina e si arrivi allo stato finale  $q_2$ ;
3. La transizione 11, 1 > 11, 1, 11 indica che partendo dallo stato  $q_1$  e leggendo il carattere  $a$ , la computazione preveda che si scriva il carattere  $a$ , ci si sposti a destra con la testina e si arrivi allo stato  $q_1$ .

# Capitolo 4

## Macchina Di Turing Universale

### 4.1 Come Inserire L'Input

La Macchina di Turing universale  $M$  dispone di 3 nastri usati per la computazione, in particolar modo:

- Il primo nastro è usato per contenere la stringa che indica la funzione di transizione della macchina di Turing  $M'$  che si vuole simulare;
- Il secondo nastro contiene l'input che si vuole dare alla macchina di Turing  $M'$  che si vuole simulare;
- Il terzo nastro è usato come "nastro di lavoro" e avrà il compito di contenere lo stato corrente e il carattere che si è letto; il terzo nastro è anche usato per aggiornare l'input ricevuto per  $M'$ .

L'input che si vuole inserire per la macchina di Turing  $M'$  deve essere formato in questo modo:

1. L'input deve appartenere all'alfabeto binario (quindi deve contenere solo 0 e 1);
2. Visto che la macchina di Turing universale legge caratteri dell'alfabeto internazionale, ogni lettera è stata codificata in alfabeto unario, quindi al carattere "a" corrisponde la stringa "1", al carattere "b" corrisponde la stringa "11", al carattere "c" corrisponde la stringa "111" e così via;
3. Ogni carattere deve essere separato dagli altri caratteri dal simbolo "0";
4. Il carattere  $\diamond$  di  $M$  deve essere rappresentato con un carattere dell'alfabeto internazionale codificato in alfabeto unario;

5. La stringa deve sempre terminare con uno "0".

Per esempio la stringa "abc" viene identificata dalla stringa "101101110", la stringa "aaab" viene identificata dalla stringa "101010110".

Quindi, per ricapitolare, nel primo nastro si deve incollare la stringa che viene restituita dal programma che è stato trattato nel capitolo 3, nel secondo nastro deve essere inserita una stringa appartenente all'alfabeto binario (contenente solo 0 e 1) e il terzo nastro deve essere lasciato vuoto.

## 4.2 Spiegazione Macchina Di Turing Universale

La macchina di Turing universale  $M$  è costituita da numerosi stati:

- $q_0, q_1, q_2$  e  $q_{16}$ : come detto in precedenza, lo stato iniziale di  $M$  è  $q_0$ . Questi 4 stati hanno il compito di inizializzare il terzo nastro in modo da permettere la computazione della stringa di input. In particolar modo, questi stati permettono al terzo nastro di posizionarsi nello stato  $q'_0$  della macchina di Turing  $M'$  che si vuole simulare, permettendo di leggere il primo carattere dell'input e di posizionarsi nuovamente all'inizio della stringa presente sul secondo nastro;
- $q_3, q_{12}$ : questi due stati sono fondamentali per iniziare la computazione. Infatti, lo stato  $q_3$  ha il compito di riportare il primo nastro contenente le computazioni al simbolo iniziale (in modo da leggere tutte le computazioni), dello stato; dallo stato  $q_3$  si può passare allo stato  $q_{12}$ , che permette di riposizionarsi al primo elemento del terzo nastro (in modo da leggere correttamente lo stato in cui ci si trova e il carattere che è stato letto);
- $q_4, q_{13}, q_{14}$ : dallo stato  $q_3$  si avanza allo stato  $q_4$  quando sia il primo nastro che il terzo nastro si ritrovano al primo carattere; questo viene fatto per controllare se la computazione presente nel terzo nastro è possibile nelle computazioni di  $M'$  indicate nel primo nastro; se le computazioni non corrispondono, lo stato  $q_{13}$  riporta il terzo nastro al punto di partenza, mentre, lo stato  $q_{14}$  porta il primo nastro alla prossima computazione da confrontare. Questo ciclo continua fintantoché non viene trovata una computazione compatibile (e in quel caso si passa allo stato  $q_5$  andando ad aggiornare lo stato di  $M$ ) oppure fino a quando non terminano le computazioni disponibili (in quel caso la computazione è illegale ed  $M$  si ferma);
- $q_5, q_6, q_7$ : una volta arrivati allo stato  $q_5$  significa che esiste una transizione legale in  $M'$ , pertanto si procede con il resto della computazione; lo stato  $q_5$

ha il compito di rimuovere tutto ciò che è contenuto nel terzo nastro di  $M$  in modo da prepararsi per il cambio di stato; gli stati  $q_6$  e  $q_7$  hanno il compito di ricopiare il nuovo stato nel terzo nastro;

- $q_8, q_{15}, q_{17}, q_{32}, q_{18}, q_{19}, q_{23}, q_{30}, q_{31}, q_{33}, q_{34}, q_{35}, q_9$ : tutti questi stati hanno il compito di aggiornare il secondo nastro (che contiene l'input) andando a revisionare il valore della cella indicata nella computazione. Per fare ciò inizialmente il contenuto del secondo nastro viene conservato a partire dal carattere successivo a quello che deve essere modificato; dopodiché, viene inserito il carattere modificato nel secondo nastro e viene copiato ciò che prima era presente dopo il carattere che è stato modificato; infine, vengono riposizionati i caratteri che sono a sinistra del carattere modificato in modo da avere una sequenza di caratteri. Al fine di questa operazione, si ritorna allo stato  $q_8$  e da questo si passa a  $q_9$ ;
- $q_{10}, q_{26}, q_{27}, q_{28}$ : questi stati servono per simulare il movimento della testina che va a sinistra, in particolar modo, in questo caso si opera sul nastro due e si va a sinistra fintantoché non trova uno 0; una volta trovato uno 0, questo continua ad andare a sinistra fintantoché non trova un altro 0. Una volta trovato si sposta di una posizione a destra (e si posiziona al primo carattere della stringa);
- $q_{11}, q_{20}, q_{21}, q_{22}, q_{36}, q_{24}$ : questi stati servono per simulare il movimento della testina che va a destra, la situazione è analoga a quanto spiegato nel punto precedente;
- $q_{25}$ : dopo aver simulato il movimento della testina, la computazione ricomincia a partire da  $q_3$ . Se nel terzo nastro è presente il simbolo "\$" significa che siamo giunti in uno stato finale e quindi la computazione termina.

# Capitolo 5

## Esempi Simulazione

In seguito verranno trattati alcuni esempi per mostrare il funzionamento della macchina di Turing universale mostrata in precedenza nel capitolo 4 e costruita in *JFLAP 8.0 Beta*.

### 5.1 Esempio 1

Nell'esempio 1 vedremo che la macchina di Turing universale, data una stringa  $w$  può determinare se  $w$  appartiene al linguaggio:

$$L = \{a^n b | n > 0\}$$

Il linguaggio  $L$  è riconosciuto dalla seguente macchina di Turing:

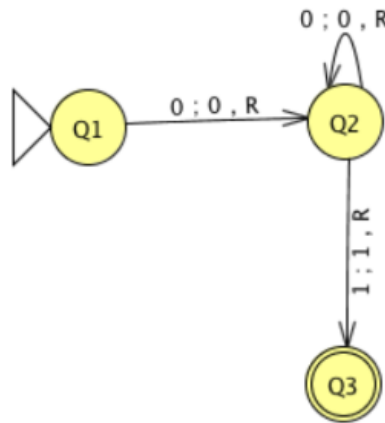


Figura 5.1: Macchina Di Turing che riconosce  $L = \{a^n b | n > 0\}$

La macchina di Turing mostrata nella figura 5.1 è rappresentata nel programma discusso nel capitolo 3 nel seguente modo:

| Codificatore UTM |       |        |        |
|------------------|-------|--------|--------|
| Stato Finale     | Stati | a      | b      |
| 0                | q0    | q1-a-r | /      |
| 0                | q1    | q1-a-r | q2-b-r |
| 1                | q2    | /      | 1      |

Figura 5.2: Rappresentazione della Macchina di Turing all'interno del programma

La stringa che si ottiene è la seguente:

$$|1, 1 > 11, 1, 11; 11, 1 > 11, 1, 11; 11, 11 > 111$, 11, 11;$$

Possiamo testare una serie di stringhe:

- La stringa " $w = aab$ " viene tradotta in alfabeto binario con 1010110, naturalmente  $w \in L$ , pertanto se la testiamo in  $M$  otteniamo messaggio che la stringa  $w$  è stata accettata;

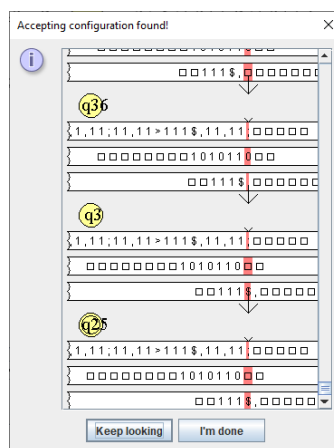


Figura 5.3: Messaggio stringa accettata per  $w = aab$

- La stringa " $w = baa$ " viene tradotta in alfabeto binario con 1101010, naturalmente  $w \notin L$ , pertanto se la testiamo in  $M$  otteniamo il messaggio che la stringa  $w$  è stata rigettata.

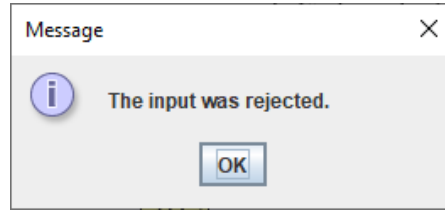


Figura 5.4: Messaggio stringa rifiutata per  $w = baa$

## 5.2 Esempio 2

Nell'esempio 2 vedremo che la macchina di Turing universale, data una stringa  $w$ , è in grado di determinare se  $w$  appartiene al linguaggio:

$$L = \{a^n b^n | n > 0\}$$

Il linguaggio  $L$  è riconosciuto dalla seguente macchina di Turing:

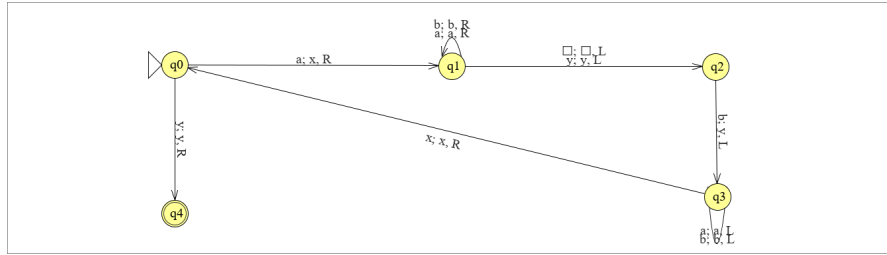


Figura 5.5: Macchina Di Turing che riconosce  $L = \{a^n b^n | n > 0\}$

La macchina di Turing mostrata nella figura 5.5 è rappresentata nel programma discusso nel capitolo 3 nel seguente modo:

| Codificatore MdT |       |        |        |        |        |        |
|------------------|-------|--------|--------|--------|--------|--------|
| Stato Finale     | Stati | c      | a      | d      | e      | b      |
| 0                | q0    | /      | q1-c-r | q4-d-r | /      | /      |
| 0                | q1    | /      | q1-a-r | q2-d-l | q2-e-l | q1-b-r |
| 0                | q2    | /      | /      | /      | /      | q3-d-l |
| 0                | q3    | q0-c-l | q3-a-l | /      | /      | q3-b-l |
| 1                | q4    | /      | /      | /      | /      | /      |

Figura 5.6: Rappresentazione della Macchina di Turing all'interno del programma

La stringa che si ottiene è la seguente:

$|1, 1 > 11, 111, 11; 1, 1111 > 11111\$, 1111, 11; 11, 1 > 11, 1, 11; 11, 1111 >$   
 $111, 1111, 1; 11, 11111 > 111, 11111, 1; 11, 11 > 11, 11, 11; 111, 11 >$   
 $1111, 1111, 1; 1111, 111 > 1, 111, 11; 1111, 1 > 1111, 1, 1; 1111, 11 > 1111, 11, 1;$

Possiamo testare una serie di stringhe:

- La stringa " $w = aaabbb$ " viene tradotta in alfabeto binario con 10101011011011011110, naturalmente  $w \in L$ , pertanto se la testiamo in  $M$  otteniamo messaggio che la stringa  $w$  è stata accettata;

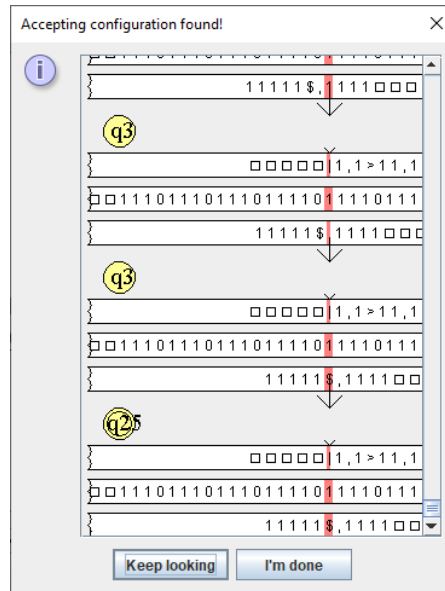


Figura 5.7: Messaggio stringa accettata per  $w = aaabbb$

- La stringa " $w = abb$ " viene tradotta in alfabeto binario con 10110110111110, naturalmente  $w \notin L$ , pertanto se la testiamo in  $M$  otteniamo il messaggio che la stringa  $w$  è stata rigettata.

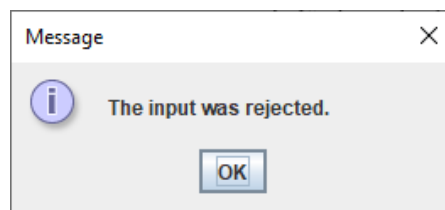


Figura 5.8: Messaggio stringa rifiutata per  $w = abb$