1. Fissato un intero $n > 0$, sia K_n il linguaggio sull'alfabeto $\{a, b\}$ denotato dall'espressione $(a+b)^* a(a+b)^{n-1} a(a+b)^*$.

   (a) A lezione abbiamo discusso come riconoscere il linguaggio K_n con un automa deterministico two-way con $O(n)$ stati. Scrivete formalmente la funzione di transizione di tale automa (rispetto al parametro n).

Solution:

$$
\begin{aligned}
\delta(b, q_0) &= (q_0, +1)\\
\delta(\lhd, q_0) &= (q_0, +1)\\
\delta(a, q_0) &= (q_{f_1}, +1)\\
\delta(*, q_{f_1}) &= (q_{f_2}, +1)\\
&\vdots\\
\delta(a, q_{f_n}) &= (q_F, +1)\\
\delta(\lhd, q_{f_n}) &= (q_{f_n}, +1)\\
\delta(b, q_{f_n}) &= (q_{b_1}, -1)\\
\delta(*, q_{b_1}) &= (q_{b_2}, -1)\\
&\vdots\\
\delta(*, q_{b_{n-2}}) &= (q_0, -1)\\
\delta(*, q_F) &= (q_F, +1)
\end{aligned}
$$

Dove gli stati q_{f_k} indicano gli n stati in cui la testina procede in avanti in cerca della a, mentre q_{b_k} indicano gli $n-1$ stati in cui la testina torna indietro.
Vediamo ad esempio che per $n = 3$ e per la parola $\rhd aabba\lhd$ abbiamo

$$
\begin{aligned}
\delta(\rhd \underline{a}abba\lhd, q_0) &= (q_{f_1}, +1)\\
\delta(\rhd a\underline{a}bba\lhd, q_{f_1}) &= (q_{f_2}, +1)\\
\delta(\rhd aa\underline{b}ba\lhd, q_{f_2}) &= (q_{f_3}, +1)\\
\delta(\rhd aab\underline{b}a\lhd, q_{f_3}) &= (q_{b_1}, -1)\\
\delta(\rhd aa\underline{b}ba\lhd, q_{b_1}) &= (q_0, -1)\\
\delta(\rhd a\underline{a}bba\lhd, q_0) &= (q_{f_1}, +1)\\
\delta(\rhd aa\underline{b}ba\lhd, q_{f_1}) &= (q_{f_2}, +1)\\
\delta(\rhd aab\underline{b}a\lhd, q_{f_2}) &= (q_{f_3}, +1)\\
\delta(\rhd aabb\underline{a}\lhd, q_{f_3}) &= (q_F, +1)\\
\delta(\rhd aabba\underline{\lhd}, q_F) &= (q_F, +1)
\end{aligned}
$$

(b) Abbiamo inoltre accennato a come riconoscere K_n con un automa sweeping con $O(n^2)$ stati, cioè un automa two way che può cambiare la direzione della testina di input solo sugli end-marker. (Nell'i-esima "passata" del nastro di input, l'automa ispezione i simboli nelle posizioni k tali che $k \equiv i \mod n$). Scrivete la funzione di transizione di tale automa nel caso $n = 3$ e generalizzatela poi a ogni n fissato.

Solution: La parte principale dell'automa sono gli n stati che si ripetono mentre si scandisce l'intera stringa. Questi devono tenere tre informazioni: due contatori da uno ad n, uno che indica la attuale iterazione sulla stringa e uno che tiene conto della distanza tra due simboli, ed infine lo scorso simbolo. Quindi abbiamo una serie di transizioni

$$\delta(a, q_{i,j,a}) = (q_{i,j+1,a}, +1)$$
$$\delta(b, q_{i,j,a}) = (q_{i,j+1,a}, +1)$$
$$\delta(a, q_{i,j,b}) = (q_{i,j+1,b}, +1)$$
$$\delta(b, q_{i,j,b}) = (q_{i,j+1,b}, +1)$$

Allo stato $q_{i,0,*}$ ci si entra attraverso uno stato q_i. Abbiamo tre casi particolari:

- abbiamo trovato due a a distanza n:

  $$\delta(a, q_{i,n,a}) = (q_F, +1)$$

  l'automa va nello stato finale

- non abbiamo trovato due a a distanza n perché ci troviamo in uno stato $q_{i,n,b}$, allora si ricomincia a contare

  $$\delta(a, q_{i,n,b}) = (q_{i,0,a}, +1)$$
  $$\delta(b, q_{i,n,b}) = (q_{i,0,b}, +1)$$

- non abbiamo trovato due a a distanza n perché il simbolo attuale è una b, allora si ricomincia a contare

  $$\delta(b, q_{i,n,a}) = (q_{i,0,b}, +1)$$

- troviamo l'end-marker di destra, abbiamo finito la stringa e cominciato a tornare indietro fino all'end-marker di sinitra:

  $$\delta(\lhd, q_{i,j,*}) = (q_{b_i}, -1)$$
  $$\delta(a, q_{b_i}) = (q_{b_i}, -1)$$
  $$\delta(b, q_{b_i}) = (q_{b_i}, -1)$$
  $$\delta(\rhd, q_{b_i}) = (q_{i,0}, -1)$$

  Nello stato $q_{i,j}$ si saltano simboli finché $j \leq i$, a quel punto si passa a $q_{i,0,*}$.

- siamo arrivati all'end-marker all'ultima iterazione:

$$\delta(\lhd, q_{n-1,j,*}) = (q_{n-1,j,*}, +1)$$

si sfora oltre perché la stringa non è accettata.

2. Modificate l'automa sweeping per K_n ottenuto nell'esercizio 1 in modo da ridurre il numero degli stati a $O(n)$.

   Suggerimento. In ciascuna passata si utilizza un contantore modulo n per individuare i simboli da ispezionare. Il contatore può essere inizializzato in funzione del valore finale del contatore nella passata precedente, evitando di contare le passate.

Solution: Si parte da uno stato q_0, e in base al simbolo iniziale si inizia il ciclo:

$$\delta(a, q_0) = (q_{1,a}, +1)$$
$$\delta(a, q_0) = (q_{1,b}, +1)$$

Ora si continua ad avanzare ciclicamente il contatore:

$$\delta(*, q_{i,*}) = (q_{i+1,*}, +1)$$

Finché non si arriva a $i = n - 1$, a questo punto come prima se il simbolo è una a, allora si ha finito, altrimenti si continua

$$\delta(b, q_{n-1,b}) = (q_{0,b}, +1)$$
$$\delta(a, q_{n-1,b}) = (q_{0,a}, +1)$$
$$\delta(b, q_{n-1,a}) = (q_{0,b}, +1)$$
$$\delta(a, q_{n-1,a}) = (q_F, +1)$$

Ora quando si arriva all'end marker $\lhd$ nello stato $q_{i,*}$, si procede a ritroso con gli stati $q_{b,j}$, con $j = 0, \ldots, n - 1$ partendo dallo stato $q_{b,i}$.

In base a con quale stato $q_{b,i}$ si arriva all'end marker di sinistra $\rhd$ si comincia con lo stato $q_{i+1,b}$, infatti i sarà l'indice con cui avremo iniziato la precedente iterazione. Questo processo va in loop se la parola non appartiene al linguaggio.

3. Svolgete l'esercizio 1 per il linguaggio formato da tutte le stringhe in cui esistono due simboli a distanza n tra loro che sono uguali.

Solution: Per ogni simbolo l'automa può andare avanti n sibmoli e vedere se è uguale altrimenti tornare indietro di $n-1$ simboli.

Alternativamente per l'automa sweeping basta modificare il comportamento dello stato $q_{n-1,b}$ per accettare il simbolo b invece che fallire a tavolino.

4. Svolgete l'esercizio 1 per il linguaggio formato da tutte le stringhe in cui tutti i simboli a distanza n tra loro sono uguali.

Solution: Questo è più semplice da fare come un automa sweeping, infatti basta controllare che tutti gli elementi nella stessa posizione $\mod n$ siano uguali.

5. Rivedere la simulazione di automi two-way deterministici mediante automi one-way deterministici presentata a lezione e studiate come possa essere modificata nel caso l'automa di partenza sia two-way nondeterministico.

Solution: Nel caso di automi non deterministici ad un certo stato p nella tabella possono essere associate più colonne ad 1 – più stati nuovi in cui può finire. Allora va cambiata la costruzione della tabella τ, iniziando da τ_ϵ vale che

$$\forall p \in Q \; \forall q \in \delta(p, \rhd) \mid \tau_\epsilon(p, q) = 1$$

Inoltre dato un $\sigma \in \Sigma, w \in \Sigma^*$, dato τ_w calcoliamo $\tau_{w\sigma}$. Ci possono essere due casi:

- $\forall p, q \in Q \mid (q, +1) \in \delta(p, \sigma) \Rightarrow \tau_{w\sigma}(p, q) = 1$
- $\forall p, r_1 \in Q \mid (r_1, -1) \in \delta(p, \sigma)$, allora, dopo un numero arbitrario di passaggi nella stringa w, torneremo al simbolo σ in degli stati $P = \{p_1', p_1'', \dots\}$, e

  $$\forall p_1 \in P \mid \tau_w(r_1, p_1) = 1$$

  Ora per ogni elemento $p_1 \in P$ ripetiamo il passo di sopra con $\delta(p_1, \sigma)$, quindi:

  - $(q, +1) \in \delta(p_1, \sigma)$, $\tau_{w\sigma}(p, q) = 1$
  - $(r_2, -1) \in \delta(p_1, \sigma)$, ripeto il processo di sopra

FINISCO !!!