

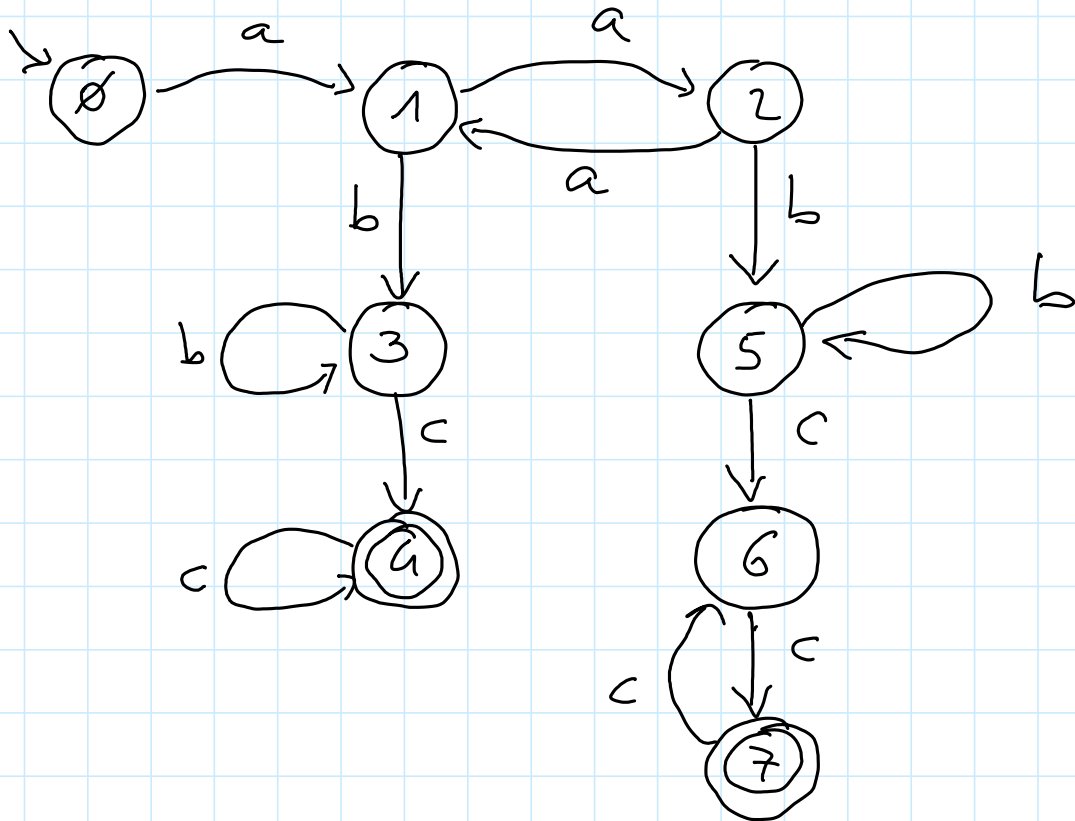
Esercizio 1)

martedì 20 agosto 2019

11:04

Il linguaggio è regolare.

Automa che lo riconosce



Grammatica che genera il linguaggio

$$S \rightarrow aaABCcc \mid aABCc \mid aABCcc$$
$$A \rightarrow \epsilon \mid aaA$$
$$B \rightarrow b \mid bB$$
$$C \rightarrow \epsilon \mid ccC$$

Esercizio 2)

martedì 20 agosto 2019

11:08

```
int conteneq (int a[], int inizio, int fine)
```

```
{ int i;  
  int c=0;  
  for (i = inizio; i < fine; i++)  
    if (a[i] < 0) c++;  
  return c;  
}
```

```
int formula (int a[], int dim)
```

```
{ int i=1;  
  int trovato = 0;  
  while (i < dim - 1 && !trovato)  
    if (conteneq(a, 0, i) == conteneq(a, i+1, dim))  
      trovato = 1;  
    else i++;  
  return trovato;  
}
```

Esercizio 3)

martedì 20 agosto 2019 11:11

let less l =

let f x (somme, ris, ult) =

if ult then (x, ris, false)

else if not ris then (somme, ris, ult)

else if x < somme

then (somme + x, ris, ult)

else (somme, false, ult)

in

let (s, r, u) = foldl f (0, true, true) l

in r ;

Esercizio 4)

martedì 20 agosto 2019 11:15

let len l =

let rec len_aux l = match l with

[] → (0, true)

| [x] → (x, true)

| x::y::ys → let (s, b) = len_aux (y::ys)

in

if b & x < s

then (s+x, b)

else (s, false)

in let (s, b) = len_aux l

in b ;;