# Esercizio 1)

lunedì 9 settembre 2019 13:35

Il linguaggio non è regolare. Si può dimostrare con il pumping lemma.

Qualunque sia $n \in \mathbb{N}^+$ prendiamo la stringa $w = a^n b^n c^{2n+1} \in L$

$|w| = 4n+1 > n$

Tutte le possibili divisioni di $w$ in $xyz$, con $|xy| \leq n$ e $y \neq \varepsilon$, possono essere rappresentate da:

$x = a^s$, $0 \leq s < n$

$y = a^t$, $0 < t \leq n-s$

$z = a^k b^n c^{2n+1}$, $k = n-s-t$.

Per $i=2$, $xy^iz \notin L$, infatti

$xy^2z = a^s a^t a^t a^k b^n c^{2n+1} =$
$a^{n+t} b^n c^{2n+1}$

Poiché $t>0$, $n+t+n \geq 2n+1$,

Poiché $t > 0$, $n + t + n \geq 2n + 1$,
quindi $a^{n+t} b^n c^{2n+1} \notin L$

---

Una grammatica che genera $L$ è la seguente

$S \rightarrow AC$

$A \rightarrow aBc \mid aAc$

$B \rightarrow bc \mid bBc$

$C \rightarrow c \mid cC$

Esercizio 2)

lunedì 9 settembre 2019 13:42

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

int num_max (int a[], int dim)
{ int i;
  int n = 0;
  int m;
  for (i = 0; i < dim; i++)
  { int aus = conta (a, i, dim, a[i]);
    if (aus > n)
      { n = aus;
        m = a[i];
      }
  }
```

```
    }
    return m;
}
```

Esercizio 3)

lunedì 9 settembre 2019 13:48

```
let max l =
  let f x (val, n, max, ult) =
    if ult then (x, 1, max, false)
      else if x = val
        then (val, n+1, max, ult)
        else if n > max
          then (x, 1, n, ult)
          else (x, 1, max, ult)
  in let (v, n, m, u) = foldr f (0, 0, 0, true) l
    in if n > m then n
                else m ;;
```

Esercizio 4)

lunedì 9 settembre 2019 13:54

```
let max l =
    let rec max_a l n max =
        match l with
        [] -> max
        | [x] -> if n > max then n else max
        | x :: y :: ys when x = y -> max_a (y::ys) (n+1) max
        | x :: y :: ys when x <> y ->
            if n > max then max_a (y::ys) 1 n
                        else max_a (y::ys) 1 max

    in max_a l 1 0 ;;
```