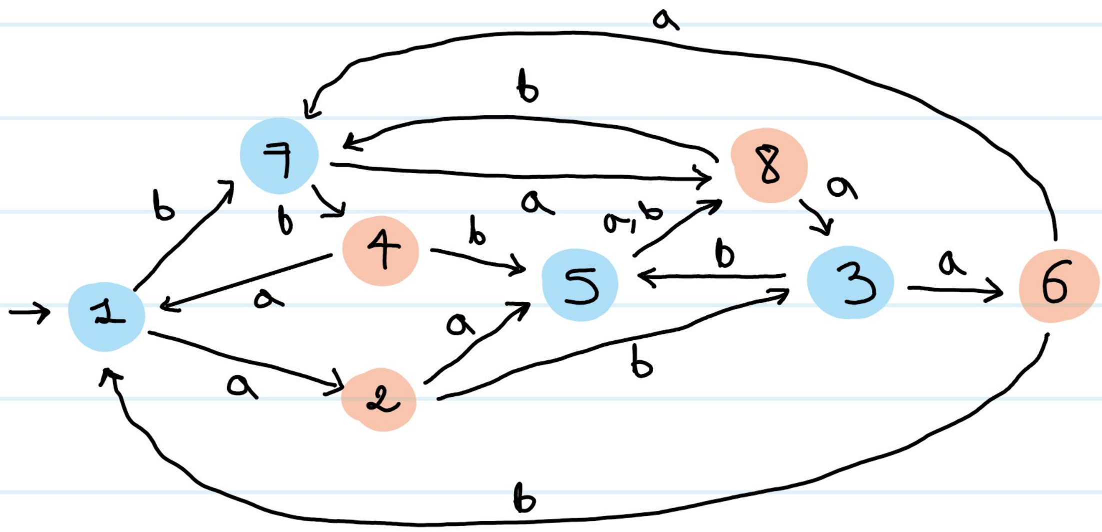
es. 1.2021



(i) $\{1,3,5,7\}$ $\{2,4,6,8\}$

a: $\{1,3,5,7\}$ $\{2,4,6,8\}$

b: $\{1,3\}$ $\{2,4,6,8\}$

$\{5,7\}$

$\{1,3\}$ $\{5,7\}$ $\{2,4,6,8\}$

a: $\{1,3\}$ $\{5,7\}$ $\{2,6\}$ $\{4,8\}$

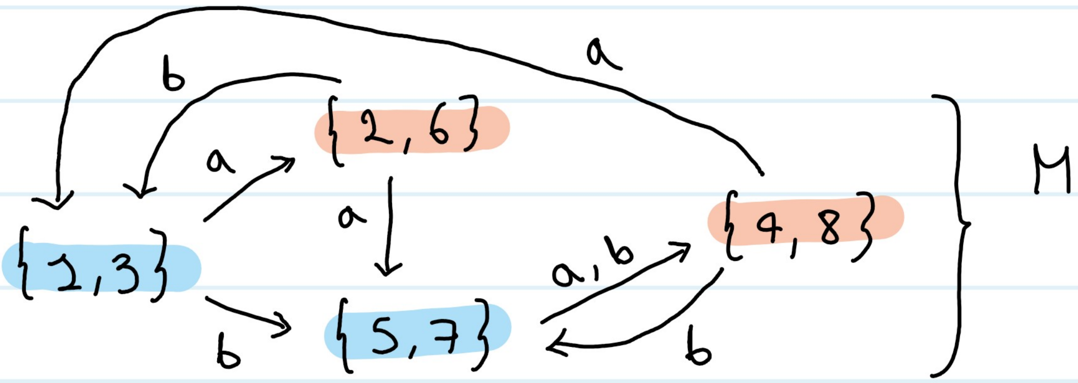
b: $\{1,3\}$ $\{5,7\}$ $\{2,6\}$ $\{4,8\}$

$\{1,3\}$ $\{5,7\}$ $\{2,6\}$ $\{4,8\}$

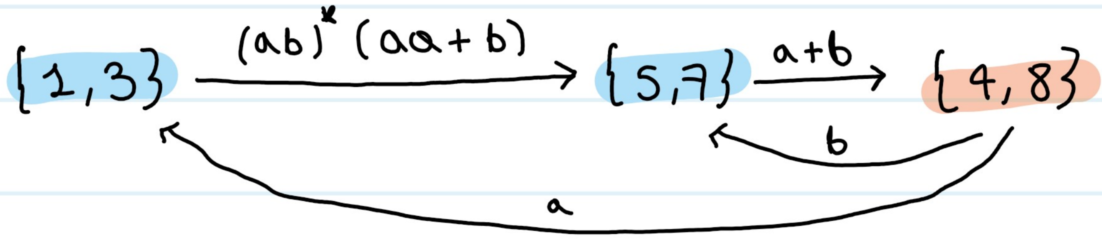
a: $\{1,3\}$ $\{5,7\}$ $\{2,6\}$ $\{4,8\}$

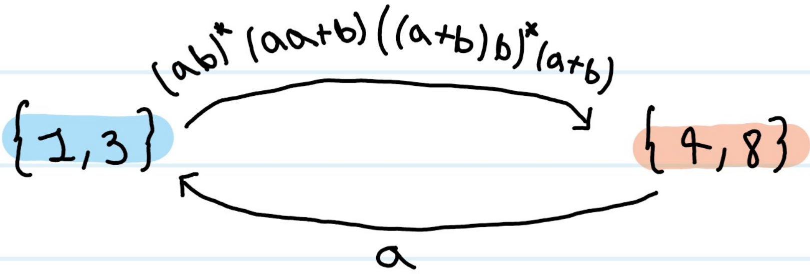
b: $\{1,3\}$ $\{5,7\}$ $\{2,6\}$ $\{4,8\}$

$$Q/\!\sim \; = \{\{1,3\}, \{5,7\}, \{2,6\}, \{4,8\}\}$$



(ii)





(iii) $P = \{ \{1,3\} \rightarrow a\{2,6\} \mid b\{5,7\},\ \{2,6\} \rightarrow b\{1,3\} \mid a\{5,7\}$
$\mid \varepsilon,\ \{5,7\} \rightarrow a\{4,8\} \mid b\{4,8\},\ \{4,8\} \rightarrow a\{1,3\} \mid$
$\mid b\{5,7\} \mid \varepsilon \}$

$$G = (\underbrace{\{\{1,3\},\{2,6\},\{5,7\},\{4,8\}\}}_{V}, \underbrace{\{a,b\}}_{T}, P, \underbrace{\{1,3\}}_{E})$$

es. 2.2021

```c
typedef enum {False, True} boolean;

boolean check(int vet1[], unsigned int dim1,
      int vet2[], unsigned int dim2){

    if (dim1 == 0) {
        return True;
    }

    if (dim2 == 0) {
        return False;
    }

    int i=0; int j=0;
    boolean valid = True;
```

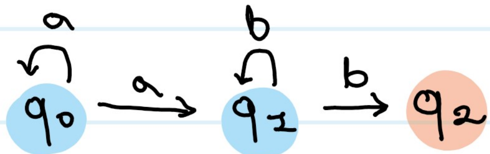
```
    boolean checked = False;

    while ( valid == True && checked == False) {
        if ( vet1[i] != vet2[J]) {
            if (J == dim2 - 1 || vet1[i] < vet2[J]) {
                valid = False;
            } else {
                J ++;
            }
        } else {
            if (i == dim1 - 1) {
                checked = True;
            } else {
                i++;
            }
        }
    }

    return valid;
}
```

es. 3. 2021

(i) $L_1 = \{ a^n b^m \mid n, m > 0 \}$



accetta L_1. Quindi L_1 è regolare, poiché regolare è anche libero.

(ii) $L_2 = \{ a^n b^m \mid 0 < n < m \}$

Si suppone L_2 sia regolare e si assume che un automa che lo accetti abbia n stati. Allora per il Pumping Lemma, $\exists\, w = xyz \mid |xy| \leq n,\ y \neq \varepsilon,\ xy^iz$ sia accettata $\forall\, i \in \mathbb{N}$. Tuttavia y è forzatamente una composizione di a: xy^mz dovrebbe essere accettata, ma violerebbe la condizione $0<n<m$, ↯.

Non soddisfacendo il Pumping Lemma, L_2 non è regolare.

$$P \begin{cases} E \rightarrow I \mid Ib \\ I \rightarrow abb \mid aIb \end{cases}$$

$G = (\{E, I\}, \{a, b\}, P, E)$ accetta come linguaggio L_2, quindi è libero.

(iii) $L_3 = \{a^n b^n c^n \mid n > 0\}$

Si suppone che L_3 sia libero e che una grammatica che lo accetti abbia n produzioni. Allora per il Pumping Lemma $\exists\, w = mnopq \mid np \neq \varepsilon$, $|nop| \leq n$, $mn^i op^i q$ sia accettata dalla grammatica.
Poiché $|nop| \leq n$, nop non può contenere contemporaneamente a e c:

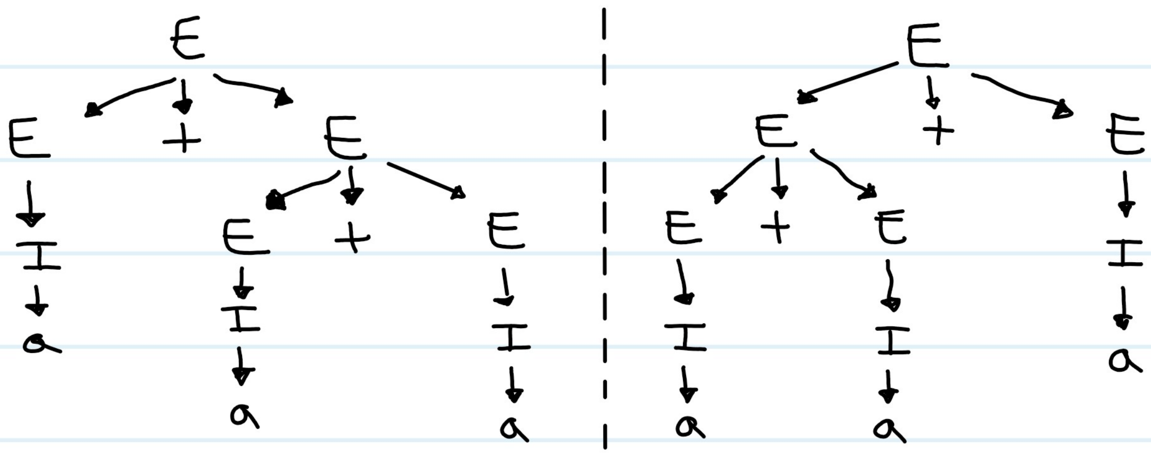
- se nop non contiene c, allora moq, che dovrebbe essere accettata, contiene più c che a e b, $\lightning$.
- se nop non contiene a, allora moq, che dovrebbe essere accettata, contiene più a che b e c, $\lightning$.

Pertanto L_3 non è libero; e poiché non è libero, non è regolare.

es. 4.2021

$$E \rightarrow I \mid E+E \mid (E)$$
$$I \rightarrow a \mid b$$

(i)



Questi due alberi producono entrambi a+a+a, quindi G é ambigua.

(ii)

$$E \rightarrow P \mid P+E$$
$$P \rightarrow a \mid b \mid (E)$$