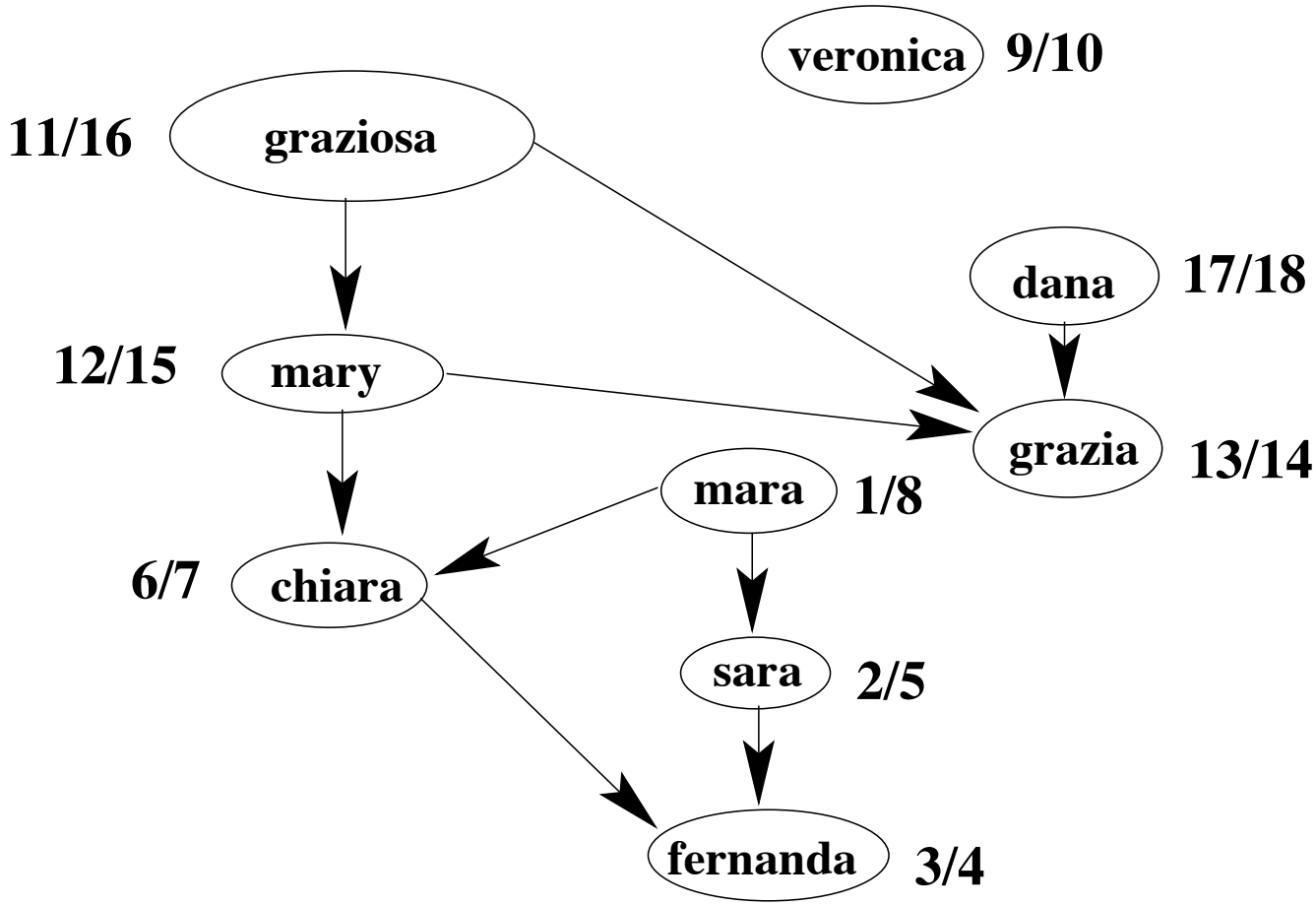
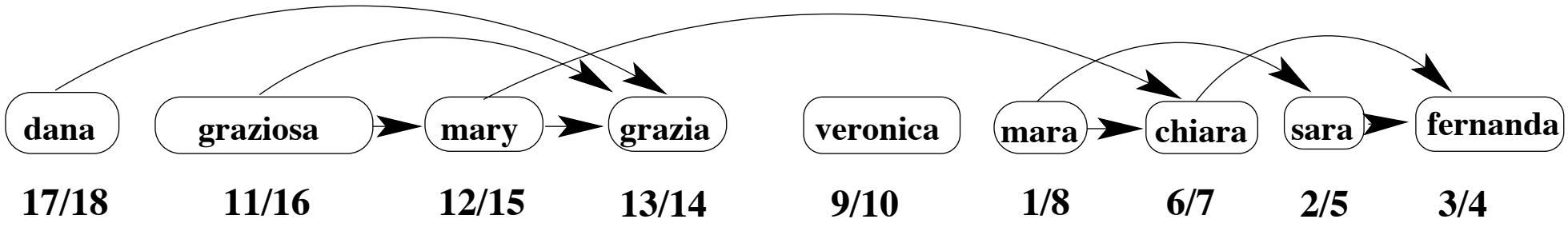
Ordinamento topologico



Ordinamento topologico



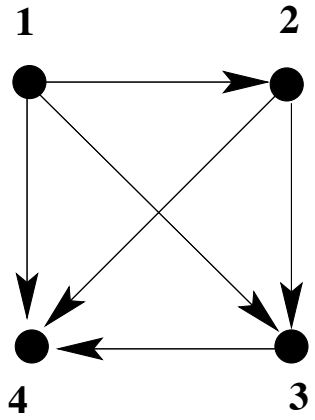
Ordinamento topologico

Un **DAG** è un **grafo diretto aciclico**. Un **ordinamento topologico** di un DAG $G = (V, E)$ è un **ordinamento totale** $\prec$ dei suoi vertici tale che:

$$\text{se } (u, v) \in E \text{ allora } u \prec v.$$

Ordinamento topologico

- Quanti ordinamenti topologici ha un grafo di n nodi privo di archi?
- Un grafo diretto è **semiconnesso** se, per ogni due nodi u e v, almeno uno dei due è raggiungibile dall'altro. Quanti ordinamenti topologici ha un DAG **semiconnesso**?



Ordinamento topologico

Teorema *Sia $G = (V, E)$ un DAG. Sia $v_1, v_2, \ldots, v_n$ una sequenza di nodi di G in ordine decrescente rispetto al tempo di fine di una visita in profondità di G. Allora $v_1, v_2, \ldots, v_n$ è un ordinamento topologico.*

Dimostrazione

Occorre dimostrare che se (v_i, v_j) è un arco di G allora $i < j$, cioè $f[v_i] > f[v_j]$. Quando l'arco (v_i, v_j) è esplorato da DFS, v_j non può essere grigio, altrimenti (v_i, v_j) sarebbe un arco back, e quindi ci sarebbe un ciclo. Dunque il colore di v_j è bianco o nero.

Se v_j è bianco, esso diventerà un discendente di v_i, e dunque, $f[v_i] > f[v_j]$. Se v_j è nero, allora esso è già stato terminato, e $f[v_j]$ è già stato assegnato. Dato che v_i non è ancora stato terminato, $f[v_i]$ deve essere ancora assegnato, e dunque $f[v_i] > f[v_j]$.

$DFSVisit(u)$

1: $color[u] \leftarrow$ GRAY
2: $time \leftarrow time + 1$
3: $d[u] \leftarrow time$
4: **for** each $v \in Adj[u]$ **do**
5: **if** $color[v] =$ WHITE **then**
6: $\pi[v] \leftarrow u$
7: $DFSVisit(v)$
8: **end if**
9: **end for**
10: $color[u] \leftarrow$ BLACK
11: $time \leftarrow time + 1$
12: $f[u] \leftarrow time$
13: $Push(S, u)$

$Topological - Sort(G)$

1: $S \leftarrow \emptyset$
2: **for** each $u \in V[G]$ **do**
3: $color[u] \leftarrow$ WHITE
4: $\pi[u] \leftarrow$ NIL
5: **end for**
6: $time \leftarrow 0$
7: **for** each $u \in V[G]$ **do**
8: **if** $color[u] =$ WHITE **then**
9: $DFSVisit(u)$
10: **end if**
11: **end for**
12: **while not** $StackEmpty(S)$ **do**
13: **print** $Pop(S)$
14: **end while**