Titolo nota 17/03/2016

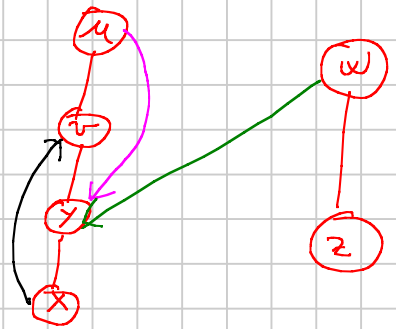
DFS(G) G è orientato — Visita di un grafo in ordine DFS

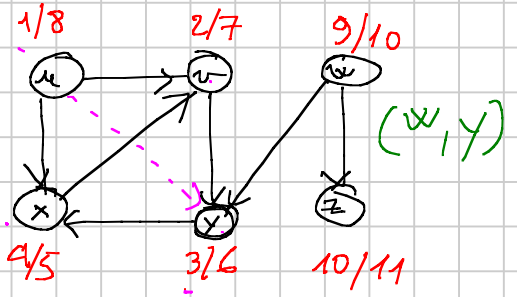
per ogni $u \in G$ {
  $u.colore = bianco$
  $u.\pi = NIL$
} $time = 0$
per ogni $u \in G$ {
  if $u.colore = bianco$ DFS_Visit(G) }

$u.\pi$ = predecessore
$u.d$ = tempo di visita
$u.f$ = tempo di fine visita

DFS_Visit(G, u):
$time = time + 1$;
$u.d = time$; $u.colore = grigio$;
per ogni $v \in Adj(u)$ {
  if (colore == bianco) { $v.\pi = u$; DFS_Visit(G, v); print (u,v) arco della foresta;
  if (colore == grigio) print (u,v) arco all'indietro;
  else if $u.d < v.d$ print (u,v) arco in avanti
  else print (u,v) arco trasversale
} $u.colore = nero$; $time = time + 1$;
$u.f = time$;

Foresta DFS





grafi orientati

Titolo nota 17/03/2016

Forest 1) archi della foresta DFS : (u,v) u è pred. di v nella DFS ;

Back 2) archi all'indietro : (u,v) : u è antenato di v ;

Forward 3) archi in avanti : (u,v) e v è antenato di u

Crossing 4) archi trasversali : (u,v) u non è antenato né discendente

} nella DFS

1) v è bianco
2) v è grigio

} grafi non orientati caso 1) e 2)

3)
4)
} v è nero

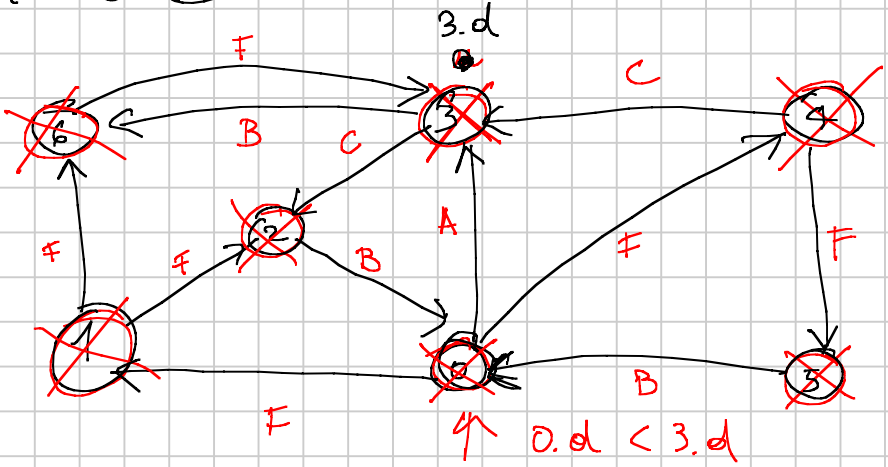
se $u.d < v.d$ allora (u,v) in avanti

se $u.d > v.d$ allora (u,v) trasversale

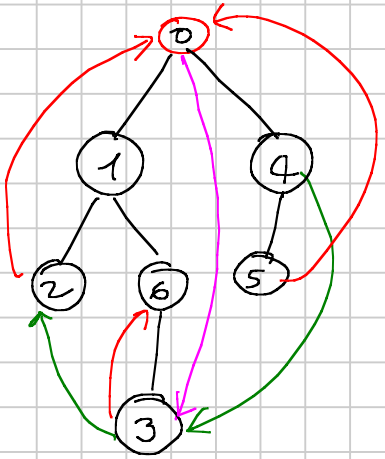
G Aciclico ? non ha archi all'indietro.

Titolo nota 17/03/2016

Contare le componenti connesse con la BFS?
di un grafo orientato
DFS che stampa gli archi e assegna loro l'etichetta



A: avanti
B: indietro
F: foreste
C: attrav.
BFS:



foreste DFS

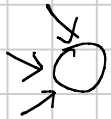
DAG

Titolo nota 17/03/2016

Directed acyclic graph

grafo orientato aciclico

non è un ciclo

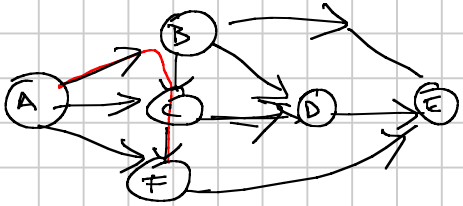


A B C F D E

Ordinamento topologico

A B C D F E

sorgente



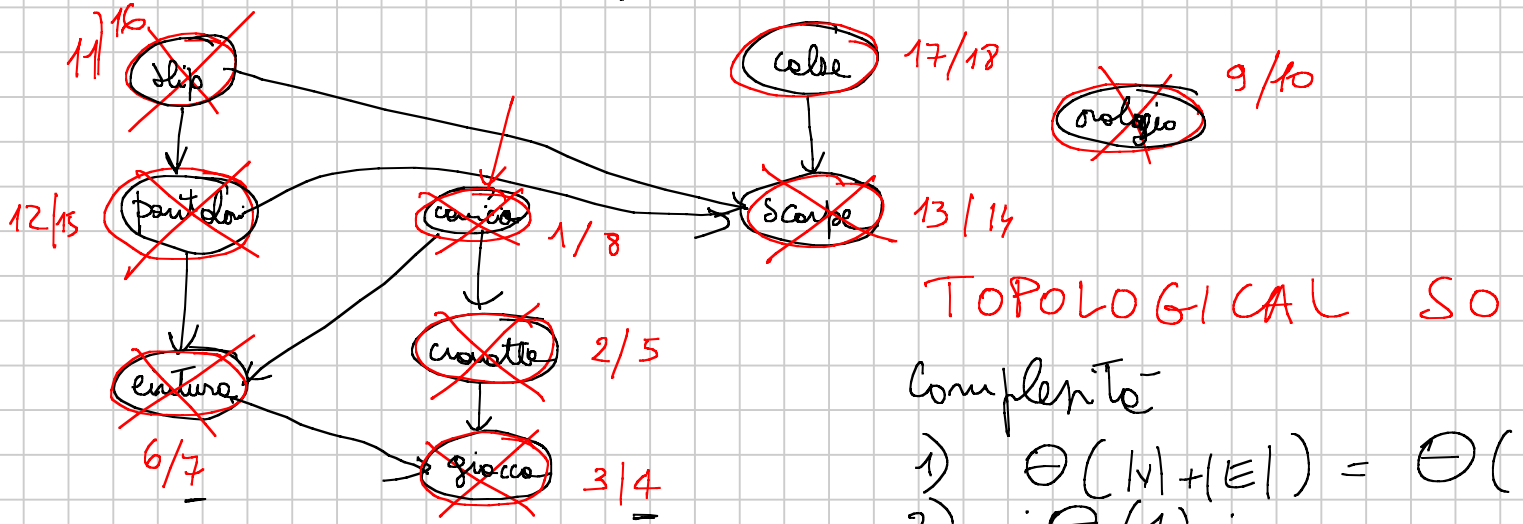
pozzo

grafi serie / parallelo

Topological Sort = ordinamento completo dei vertici

GRAFO = DAG operazioni necessarie a vestirsi

Titolo nota 17/03/2016



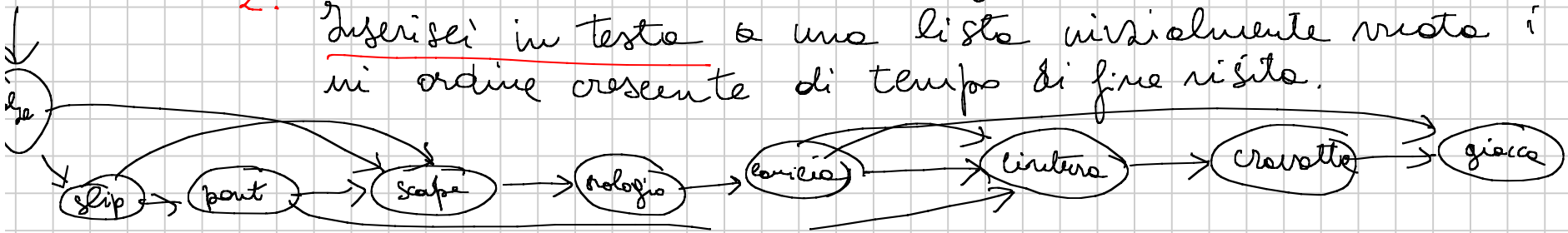
TOPOLOGICAL SORT

Complessità

1) $\Theta(|V|+|E|) = \Theta(n+m)$

2) $\Theta(1)$

1. DFS(G) e partire da una sorgente per calc. i tempi di fine visita di ciascun vertice v.f
2. Inserisci in testa a una lista inizialmente vuota i vertici in ordine crescente di tempo di fine visita.



G orientato

Titolo nota 17/03/2016

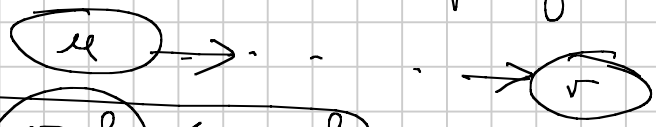
Lemma G è aciclico se non ha archi all'indietro

durante la visita non si incontrano vertici grigi

TOPOLOGICAL_SORT(G) modifica di DFS che costruisce la sequenza di ordine topologico

è corretta.

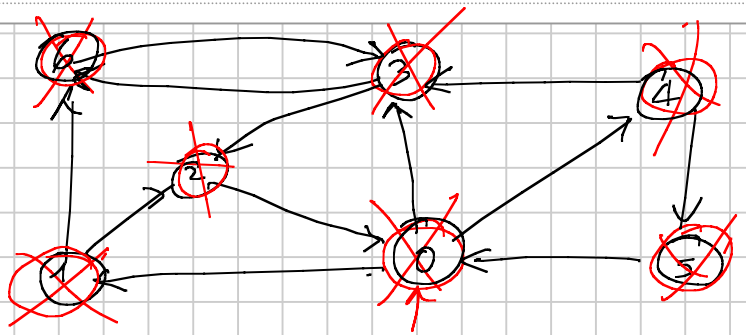
se esiste un arco $(u,v) \in G \Rightarrow v.f < u.f$

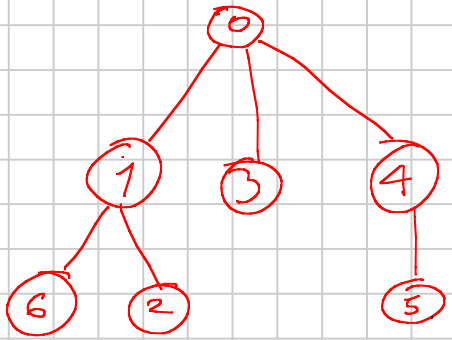


quando v è esaminato dalla DFS v è bianco o nero.

(non può essere grigio per il lemma) ↓ u predecessore di v ↓ è maggior ragione

Titolo nota 17/03/2016





Alb.

BFS

BFS: 0, 1, 3, 4, 6, 2, 5

Alb	Alb	Alb	Alb	Alb	B	B	B	Alb	B	B
(0, 1)	(0, 3)	(0, 4)	(1, 6)	(1, 2)	(3, 6)	(3, 2)	(2, 0)	(4, 5)	(4, 3)	(5, 0)