

INFORMATICA GENERALE - IV esonero

docente: IVANO SALVO, 12/1/2024

Risolvere **uno solo** dei seguenti problemi.

Problema 1: Dato un grafo non orientato $G = (V, E)$ e un insieme $C \subseteq V$ di nodi detti *centri*, diciamo che $v \in V$ appartiene alla *sfera di influenza* $\mathbf{S}(c)$ di $c \in C$ se e solo se $d(c, v) < d(c', v)$ per ogni $c' \in C$ ($c' \neq c$). Viceversa, un nodo v è detto di *frontiera*, se esistono almeno due nodi $c, c' \in C$ tali che $d(c, v) = d(c', v)$.

1. Dato il grafo in Fig. 1, determinare le sfere d'influenza dei centri (nodi grigi) e i nodi di frontiera;
2. disegnare un grafo connesso con un insieme $C \subseteq V$ di centri tali che: - $|C| = 3$; - per ogni $c \in C$, $|\mathbf{S}(c)| > 1$; - e non ci sia nessun nodo di frontiera;
3. proporre una codifica per classificare i nodi in sfere d'influenza e in nodi di frontiera. Dare un algoritmo che dato un grafo $G = (V, E)$ e un insieme di centri $C \subseteq V$, calcola la classificazione dei nodi di V nella codifica scelta. Dare la complessità in tempo/spazio in funzione di $m = |E|$, $n = |V|$ e $k = |C|$.

Problema 2: In un grafo orientato *finito* c'è un insieme I di nodi *iniziali* (grigio chiari in Fig. 2) e un insieme F di nodi *finali* (grigio scuri in figura).

Consideriamo le proprietà: - (**E**) esiste in G una passeggiata *infinita* che parte da un nodo iniziale e passa infinite volte su un nodo finale; - (**A**) ogni passeggiata *infinita* in G che parte da un nodo iniziale passa infinite volte su un nodo finale.

1. In un DAG può valere (**E**)? Ed (**A**)?
2. Dire perché nel grafo in Fig. 2 vale (**E**) ma non vale (**A**). Rendere finale un nodo e togliere un arco affinché valga (**A**).
3. Determinare delle condizioni affinché valga (**E**) e delle condizioni per (**A**).
4. Descrivere un algoritmo che dati un grafo G , e $I, F \subseteq V$, determina se è soddisfatta la proprietà (**E**). Modificare lo stesso algoritmo per verificare (**A**).

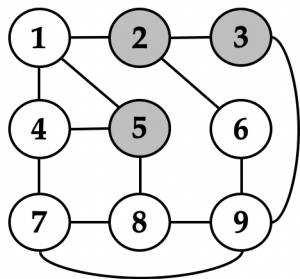


Figura 1: Esempio Problema 1

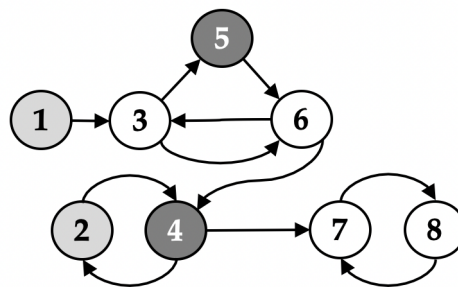


Figura 2: Esempio Problema 2