

ASD Laboratorio 04

Antonio Bucchiarone/Marta Fornasier/Cristian Consonni/
Francesco Lotito/Giovanni Zotta

UniTN

2020-11-17

06/10	Introduzione
13/10	Ad-hoc
10/11	Grafi 1
17/11	Grafi 2
10/12	Progetto 1
15/12	Progetto 1

Progetto:

- Dal 9 al 16 dicembre (consegna ore 18:00);
- Iscrizione dei gruppi al progetto entro il **4 dicembre**:
http://bit.ly/ASDprog_2020-2021
(dovete essere loggati con l'account UniTN)

- Basta una qualunque visita (in ampiezza, in profondità, ecc...)
- Implementazioni sul sito:

https://judge.science.unitn.it/slides/asd20/soluzioni_lab03.zip.

IMPLEMENTAZIONE DFS RICORSIVA - INPUT

```
struct nodo{
    vector<int> vic;
    bool visited = false;
};
...
vector<nodo> grafo;
...
in >> N;
grafo.resize(N);
for(int i=0;i<M;i++){
    int from, to;
    in >> from >> to;
    grafo[from].vic.push_back(to);
}
```

IMPLEMENTAZIONE DFS RICORSIVA

Con grafo variabile globale (*caveat emptor*):

```
void visit(int n){
    grafo[n].visited=true;
    count++;
    for(int v:grafo[n].vic)
        if(!grafo[v].visited)
            visit(v);
}
```

Passando grafo per reference:

```
void visit(int n, vector<nodo>& grafo, int& count) {
    grafo[n].visited = true;
    count++;
    for(int v:grafo[n].vic)
        if(!grafo[v].visited)
            visit(v, grafo);
}
```

IMPLEMENTAZIONE DFS CON STACK

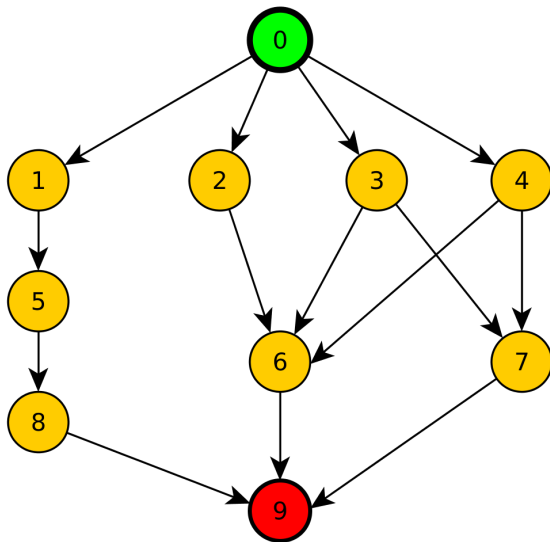
```
stack<int> s;  
s.push(S);  
while(!s.empty()){  
    int n = s.top();  
    s.pop();  
    // se un nodo è già stato visitato lo ignoriamo!  
    if(grafo[n].visited) { continue; }  
    grafo[n].visited = true;  
    count++;  
    // visita tutti i vicini  
    for(int v: grafo[n].adj) {  
        s.push(v);  
    }  
}
```

IMPLEMENTAZIONE BFS CON CODA

```
queue<int> q;
q.push(S);
while(!q.empty()) {
    int n = q.front();
    q.pop();
    // se un nodo è già stato visitato lo ignoriamo!
    if(grafo[n].visited) { continue; }
    grafo[n].visited = true;
    count++;
    // visita tutti i vicini
    for(int v: grafo[n].adj) {
        q.push(v);
    }
}
```

- Per ogni nodo di partenza, trovare il nodo più lontano.
- La distanza in un grafo non pesato si può calcolare utilizzando una BFS.

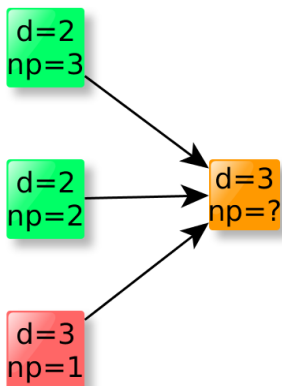
ESEMPIO: CAMMINI MINIMI



SOLUZIONE NUMERO DI CAMMINI MINIMI

IDEA

- Stiamo calcolando il numero di diversi cammini minimi da s a t
- Sia $P(v)$ l'insieme di predecessori di v , ovvero di tutti i w tali che:
 - ▶ $(w, v) \in E$
 - ▶ $d(s, v) = d(s, w) + 1$
- $NUMPATH(v) = \sum_{w \in P(v)} NUMPATH(w)$
- Calcoliamo $NUMPATH(t)$



ESERCIZI (I)

DIMENSIONE MASSIMA COMPONENTE FORTEMENTE CONNESSA

Dato un grafo orientato trovare la **dimensione** della massima componente fortemente connessa.

ORDINAMENTO TOPOLOGICO

Dato un grafo diretto aciclico, trovare un suo ordinamento topologico.

CAMMINO PIÙ LUNGO

Dato un grafo diretto aciclico, trovare la lunghezza del suo cammino più lungo.

ESERCIZI (II)

CICLI CICLABILI

Primo progetto dell'a.a. 2019/2020

BATMAN SI ANNOIA

Primo progetto dell'a.a. 2014/2015