



HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Suunnatut verkot

Algoritmit ongelmanratkaisussa,
kevät 2018





Oheismateriaalia

- Kisakoodarin käsikirja, luku 16
- <https://cses.fi/kkkk.pdf>



Verkon tallentaminen

- yleensä hyvä tapa: taulukko, jossa on vieruslistoja
- esimerkki: verkossa on solmut $1, 2, \dots, 10$:

```
vector<int> graph[10+1];
```

- kaaren $2 \rightarrow 5$ lisääminen verkkoon:

```
graph[2].push_back(5);
```



Syvyyshaku

- syvyyshaku on perusmenetelmä verkon läpikäyntiin
- mukava rekursiivinen toteutus:

```
void dfs(int s) {  
    if (visited[s]) return;  
    visited[s] = true;  
    for (auto u : graph[s]) dfs(u);  
}
```



Topologinen järjestäminen

- jos verkko on suunnattu ja syklitön, voimme aina muodostaa topologisen järjestyksen
- kaari $a \rightarrow b$ tarkoittaa, että solmun a täytyy tulla ennen solmua b järjestyksessä
- on monta tapaa etsiä järjestys lineaarisessa ajassa, kirjassa kerrotaan niistä yksi



Dynaaminen ohjelmointi

- voimme käyttää myös dynaamista ohjelmointia topologisessa järjestyksessä
- pystymme ratkaisemaan tehokkaasti monia ongelmia, jotka olisivat vaikeita yleisessä verkossa



Funktionaaliset verkot

- funktionaalinen verkko on suunnattu verkko, jonka jokaisesta solmusta lähtee tasan yksi kaari
- funktion arvo $f(a) = b$ tarkoittaa, että solmusta a on kaari solmuun b
- voimme käsitellä funktionaalisia verkkoja puiden tapaan ja hyödyntää esimerkiksi tehokasta etenemistä