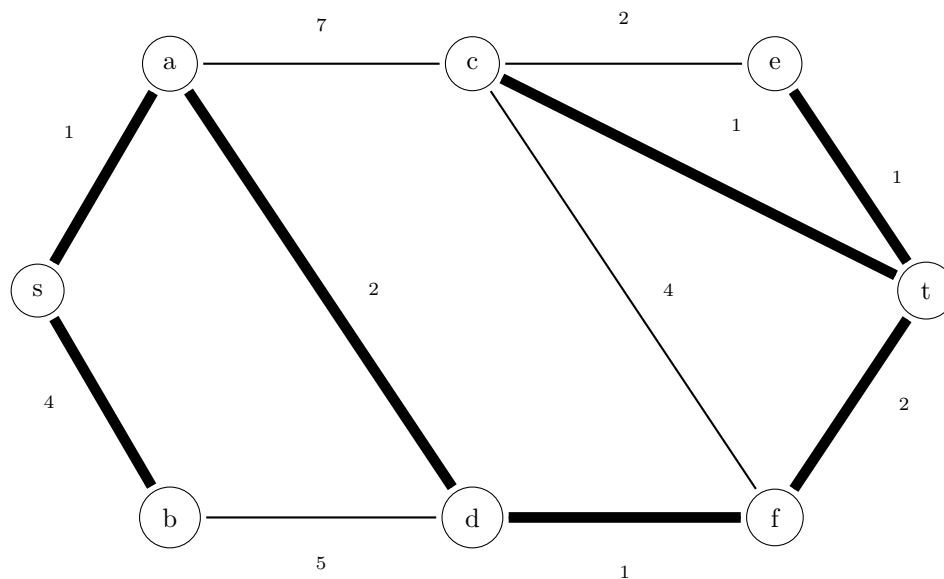


Kodutöö esitamise tähtaeg: 17. detsember, 23:59

## Minimaalse kaaluga toesuspuu

Olgu  $G = (V, E)$  sidus ja kaalutud servadega graaf. Graafi  $G$  toesuspuuks nimetatakse puud  $T = (V, E')$ , kus  $E' \subseteq E$ . Teisiti öeldes on toesuspuu  $T$  selline graafi  $G$  alamgraaf, mis ühendab kõiki tema tippe, kuid ei sisalda tsükleid. Igal sidusal graafil leidub toesuspuu.

Graafi  $G$  minimaalseks toesuspuuks nimetatakse sellist toesuspuud, mille servade kaalude summa on vähim. On teada mitmeid efektiivseid algoritme toesuspuu leidmiseks. Selles praktikumis vaatleme kahte ahnet algoritmi: Primi algoritmi ja Kruskali algoritmi.



Joonis 1: Paksemad jooned tähistavad antud graafi minimaalset aluspuud.

**Ülesanne 1 (75%)** Implementeerida minimaalse toesuspuu leidmine kas Primi või Kruskali algoritmiga vastavalt liidesele *MinimumSpanningTreeAlgorithm*. 15% punktidest anname graafi implementatsiooni eest.

## Rändkaupmehe ülesanne

Olgu meile antud teatud linnad ja lühimad teed iga kahe linna vahel. Rändkaupmehe ülesanne küsib, milline on lühim teekond, mis läbib kõiki linna ja jõuab lõpuks algpunkti tagasi.

Probleemi sõnastus formaalsemalt võiks olla selline: *olgu meile antud täisgraaf (iga tipp on ühendatud iga teise tipuga)  $G$ , mille servad on kaalutud. Leida vähima kaaluga tsükel, mis sisaldab kõiki tippe.*

Seni ei ole teada ühtegi efektiivset algoritmi, mis suudaks suvalise täisgraafi korral anda sellele probleemile täpse vastuse. (Efektiivse algoritmi leidmine lahendaks ka ühe olulisema küsimuse informaatikas. Vaata:

$P$  vs  $NP$ ). Teada on aga mitmeid algoritme, mis annavad ligikaudseid tulemusi. Näiteks on võimalik leida minimaalne toesuspuu ja see eesjärjestus süvitsi läbida.

**Ülesanne 2 (25%)**

Leida lähend rändkaupmehe ülesandele kasutades minimaalse kaaluga toesuspuud. Implementeerida vastavalt liidesele *ApproximateTravelingSalesman*.

Liidesed on kättesaadavad aadressil <https://github.com/ut-aa/aa2016-lab8>.