Algoritmien teoria, kevät 1999
Harjoitus 4, 15.2. klo 10–12 sali MaD380

1. Millainen puurakenne syntyy, kun luennolla (ss. 172–173) esitettyä tasapainottavaa ja tiivistävää *union-find*-algoritmia sovelletaan seuraavan ohjelman tuottamaan *union-find*-operaatiojonoon? Aluksi on $S_i = \{i\}, i = 1, \ldots, 16$.

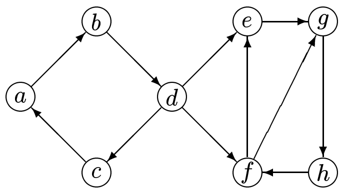
   begin
       for $i = 1$ **to** 15 **by** 2 **do** $union(i, i+1, i)$;
       for $i = 1$ **to** 13 **by** 4 **do** $union(i, i+2, i)$;
       $union(1, 5, 1)$;
       $union(9, 13, 9)$;
       $union(1, 9, 1)$;
       for $i = 1$ **to** 13 **by** 4 **do** $\mathit{find}(i)$
   end.

2. Osoita, että jos *union-find*-algoritmissa käytetään puiden tasapainotusta, mutta ei polun tiivistystä, n:n *union-find*-operaation jonon suorittaminen voi vaatia ajan $\Omega(n \log n)$. (*Vihje:* Tarkastele jonoja, joista syntyy edellisen tehtävän tapaan binomipuita.)

3. Tarkastellaan muotoa
$$X_i = Y_i, \quad i = 1, 2, \ldots, k,$$
olevien yhtälöryhmien ratkaisemista, missä kukin X_i ja Y_i on muuttuja tai kokonaisluku. Esimerkiksi ryhmällä
$$\begin{aligned} X &= 3 \\ X &= Y \end{aligned}$$
on ratkaisu $X = Y = 3$, mutta laajennetulla ryhmällä
$$\begin{aligned} X &= 3 \\ X &= Y \\ Y &= 4 \end{aligned}$$
ei ole ratkaisua. Selitä, miten *union-find*-algoritmilla voidaan tehokkaasti päättää, onko annetulla yhtälöryhmällä ratkaisua vai ei.

4. Todista seuraava väite (luentojen s. 180, Lause 2): Jos (v, w) on syvyyshaun määrittämä suunnatun verkon sivuttaiskaari, niin $\mathit{postnum}[v] > \mathit{postnum}[w]$.

5. Määritä alla olevan verkon vahvasti yhtenäiset komponentit luentojen s. 181 algoritmilla. Oletetaan, että solmut ovat vieruslistoissa aakkosjärjestyksessä ja että algoritmi aloittaa verkon tutkimisen solmusta a.



6. Sanotaan, että suunnatun verkon $G = (V, E)$ solmu u on *alkusolmu*, jos verkon kaikki muut solmut voidaan saavuttaa u:sta, so. jos kuhunkin muuhun solmuun johtaa u:sta suunnattu polku. Suunnittele tehokas algoritmi, joka etsii vieruslistoina esitetyn suunnatun verkon jonkin alkusolmun, mikäli verkossa sellaisia on. Miten muuttaisit algoritmia, jos tehtävänä olisi määrittää verkon *kaikki* alkusolmut?