Øvelser vedrørende MapReduce

Øvelse 1

a) Beskriv **map** og **reduce** funktioner der kan bruges til at transformere en liste af n par

$$(x_1,0), (x_2,0), ..., (x_n,0)$$

til listen

$$(x_1,n), (x_2,n), ..., (x_n,n)$$

dvs. tilknytter til hvert x_i det totale antal elementer i listen.

b) Beskriv **map** og **reduce** funktioner der kan bruges til at transformere en liste af n par

$$(x_1,0), (x_2,0), ..., (x_n,0)$$

til listen

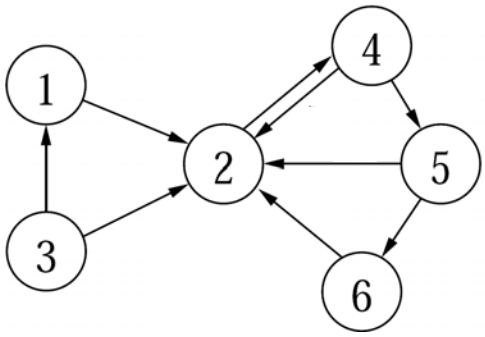
$$(x_1,k), (x_2,k), ..., (x_n,k)$$

hvor $k \leq n$ er antal **forskellige** x_i i listen.

Øvelse 2 (Afleveringsopgave)

I denne opgave vil vi se på hvordan man kan anvende MapReduce interfacet til at beregne PageRank værdierne for en webgraf. Vi antager at inputtet er givet ved en liste af henvisninger (i,j), som angiver at side i henviser til side j, og hvor både i og j antages at være heltal. For nedenstående graf har vi f.eks. følgende liste som input:

(1,2), (5,6), (2,4), (3,1), (4,2), (4,5), (6,2), (5,2), (3,2)



Vi ønsker at beregne sandsynlighedsfordelingen for at være på siderne efter s=50 skridt af RandomSurfer algoritmen, d.v.s. vi ønsker at beregne listen

$$(i_1, p_{i_1}^{(s)}), (i_2, p_{i_2}^{(s)}), (i_3, p_{i_3}^{(s)}), \ldots, (i_n, p_{i_n}^{(s)})$$

hvor $i_1, i_2, \ldots, i_n$ er de n forskellige sider der indgår i henvisningerne. Sandsynlighederne beregnes ud fra følgende formel:

$$p_1^{(0)} = 1.0 \qquad p_2^{(0)} = \cdots = p_n^{(0)} = 0.0 \qquad p_i^{(s)} = 0.85 \cdot \sum_{j:j\to i} \frac{p_j^{(s-1)}}{\text{udgrad}(j)} + 0.15 \cdot \frac{1}{n}$$

For ovenstående eksempel graf bliver dette

(1, 0.03563), (2, 0.35462), (3, 0.02500), (4, 0.32643), (5, 0.16373), (6, 0.09459)

(værdierne er beregnet v.h.a. regnearket fra PageRank øvelse 2 ved at sætte sandsynligheden til 15%).

Vi kan beregne den ønskede liste ved at foretage nedenstående transformationer på vores input liste, hvor ①, ②, og ③ udføres præcis én gang, og ④ udføres s gange. ① udvider hver henvisning med information om det totale antal sider n repræsenteret i input, ② udvider yderligere hver henvisning (i,j) med information om udgraden af i i webgrafen, og endeligt udvider ③ hver henvisning (i,j) med sandsynligheden for at stå på side i i starten af RandomSurfer processen. ④ beregner for en henvisning (i,j) den nye sandsynlighed for at stå på side i hvis vi laver et yderligere skridt i RandomSurfer algoritmen.

$(i_1, j_1), (i_2, j_2), \ldots$

① → $(i_1, j_1, n), (i_2, j_2, n), \ldots$

② → $(i_1, j_1, n, \text{udgrad}(i_1)), (i_2, j_2, n, \text{udgrad}(i_2)), \ldots$

③ → $(i_1, j_1, n, \text{udgrad}(i_1), p_{i_1}^{(0)}), (i_2, j_2, n, \text{udgrad}(i_2), p_{i_2}^{(0)}), \ldots$

④ → $(i_1, j_1, n, \text{udgrad}(i_1), p_{i_1}^{(1)}), (i_2, j_2, n, \text{udgrad}(i_2), p_{i_2}^{(1)}), \ldots$

④ → $(i_1, j_1, n, \text{udgrad}(i_1), p_{i_1}^{(2)}), (i_2, j_2, n, \text{udgrad}(i_2), p_{i_2}^{(2)}), \ldots$

...

④ → $(i_1, j_1, n, \text{udgrad}(i_1), p_{i_1}^{(s)}), (i_2, j_2, n, \text{udgrad}(i_2), p_{i_2}^{(s)}), \ldots$

a) Beskriv hvordan **mindst to** af de fire transformationer ①, ②, ③ og ④ kan implementeres v.h.a. MapReduce interfacet for passende valg af **map** og **reduce** funktioner.