4. eginkizuna

Bi funtzionalitate berri inplementatu nahi ditugu:

a) `HashMap<String, Double> pageRank()`
`//POST: emaitza web-orri zerrendaren web-orri bakoitzaren PageRank algoritmoaren balioa da`

PageRank algoritmoa (https://eu.wikipedia.org/wiki/PageRank) grafo baten dokumentuei (edo web-orriei) bere garrantzia adierazten duen zenbakizko balio bat esleitzen dien algoritmo bat da. PageRank sistema Google bilatzaileak erabiltzen du web-orri baten garrantzia erabakitzeko. PageRank-ek pertsona batek grafoa atzi eta estekak jarraituz nodo zehatz batetara iristeko duen probabilitatea kalkulatzen du. PageRank algoritmoak zenbait iterazioen beharra dauka, balio hurbildua doitzeko. Hasiera batean, nodo guztiek probabilitatea berbera dute:

$\forall\ A \in nodoak\ ,\quad PR(A)=\frac{1}{N}$,non N grafoaren nodo kopurua den.

Gero, iterazio bakoitzean, nodo bakoitzaren balioa birkalkulatu behar da honako formulaerabiliz:

$$PR(A)=\frac{1-d}{N}+d*\sum_{i=1}^{n}\frac{PR(i)}{C(i)}$$ non

- **PR(A)** A orriaren PageRank da.
- **d** (damping factor) indargetze faktorea da, 0 eta 1 arteko balioa duena (balio tipiko bat 0.85 da)
- **PR(i)** A estekatzen duen i orri bakoitzaren PageRank balioa (aurreko iteraziokoa).
- **C(i)** i orritik ateratzen diren esteka kopurua (bai A orrialdera, bat beste orrialdetara).

Prozesuak jarraituko du iterazio baten eta hurrengo iterazioaren kenketen balio absolutuen batuketa aurredefinitutako atalase bat baino txikiagoa izan arte (adibidez 0.0001).

Adibidez, demagun lau nodoez osatutako grafoa dugula : A, B, C eta D nodoak. B nodoak estekak ditu C eta A nodoetara, C nodoak esteka bat du A nodora, eta D nodoak estekak ditu A, B eta C nodoetara.

Hasieran nodo bakoitzaren balioa 0.25 izango da. Lehenengo iterazioan, B nodoak bere balioaren erdia, 0.125, pasatuko dio A nodoari, eta beste erdia C nodoari. C nodoak bere balio osoa, 0.25, pasako dio A nodoari. D nodoak hiru esteka dituenez, bere balioaren herena (gutxi gorabehera 0.083) pasako die A, B eta C nodoei. Formula aplikatu ostean, lehenengo iterazioa bukatzerakoan, A nodoaren PageRank balioa 0.427 izango da gutxi gorabehera.

```
b) public class Bikote {

      String web;

      Double pageRank;

   }

   ArrayList<Bikote> bilatzailea(String gakoHitz)

   // Post: Emaitza emandako gako-hitza duten web-orrien zerrenda da, bere
   pagerank-aren arabera handienetik txikienera ordenatuta (hau da,
   lehenengo posizioetan pagerank handiena duten web-orriak agertuko dira)



c) (aukerazkoa) ArrayList<Bikote> bilatzailea(String gakoHitz1,

                                                 String gakoHitz2)

   // Post: Emaitza emandako gako-hitzak dituzten web-orrien zerrenda da,
   bere pagerank-aren arabera handienetik txikienera ordenatuta (hau da,
   lehenengo posizioetan pagerank handiena duten web-orriak agertuko dira)
```

Honakoa entregatuko da (Astelehena, 21-XII-2020, laugarren eginkizuna entregatzeko azken data):

- Eskatutakoa exekutatzen duten programak (zuzen exekutatu behar dira). **Frogatu egin beharko da programak ondo funtzionatzen duela datu ez tribialekin (hau da, hasierako fitxategiko milaka lerro prozesatzen).**
- Eskatutako metodoen exekuzio-adibideak, proba-datuak eta emaitzak nolakoak diren azalpena. Interesgarria denean, proba bakoitzaren exekuziodenbora ere emango da.
- Dokumentazioa, emandako problema, aztertutako aukerak, inplementazioa, eraginkortasuna etabar deskribatzen duena.

Gainera Checklist-a bete eta entregatu beharko duzue, eskatutako guztia egin duzuela egiaztatzeko.

OHARRA: eskatutako emaitza batzuek konputazio-kostu altua dutenez, horrek algoritmoak entregatze-data baino lehenago funtzionatzea eskatuko du, bestela ezin izango direlako emaitzak entregatu.

<u>LAGUNTZA: iterazio bakoitzeko diferentzia</u> (iterazio bakoitzaren denbora gutxi gora behera 8 segundu)

iterazioa:	0	diff:	0.782973717764194
iterazioa:	1	diff:	0.344276535594372
iterazioa:	2	diff:	0.163840596038833
iterazioa:	3	diff:	0.087185768116801
iterazioa:	4	diff:	0.050037776206004
iterazioa:	5	diff:	0.031404833276485
iterazioa:	6	diff:	0.020827658897577
iterazioa:	7	diff:	0.014681098383163
iterazioa:	8	diff:	0.010699860422163
iterazioa:	9	diff:	0.008041772386592
iterazioa:	10	diff:	0.006132009166671
iterazioa:	11	diff:	0.004749528325397
iterazioa:	12	diff:	0.00370432722091
iterazioa:	13	diff:	0.002914058287575
iterazioa:	14	diff:	0.002300827614045
iterazioa:	15	diff:	0.001826896845737
iterazioa:	16	diff:	0.001454288501982
iterazioa:	17	diff:	0.001163004906129
iterazioa:	18	diff:	9.32E-04
iterazioa:	19	diff:	7.50E-04
iterazioa:	20	diff:	6.04E-04
iterazioa:	21	diff:	4.88E-04
iterazioa:	22	diff:	3.96E-04
iterazioa:	23	diff:	3.21E-04
iterazioa:	24	diff:	2.62E-04
iterazioa:	25	diff:	2.14E-04
iterazioa:	26	diff:	1.75E-04
iterazioa:	27	diff:	1.43E-04
iterazioa:	28	diff:	1.17E-04
iterazioa:	29	diff:	9.67E-05

iterazio kopurua: 30