

Hainbat proba egin ditut, eta proba horiek gauzatzeko erabili dudan kodea iruzkin bezala utzi dut aplikazioan (*//* erabiliz). *System.nanoTime()* erabiliz kalkulatu ditut denbora guztiak. Proba horien ondorioak hauek dira:

Hasteko, hiztegia kargatzerakoan, kargatzen diren hitzak mugatuz probak egin ditut, eta hauek izan dira emaitzak:

- Hitz guztiak kargatuz gero, aplikazioak 2.75 s behar du hiztegi osoa gordetzeko.
- 10 balioa duten hitzak kenduz gero, 1 s behar du.
- 9 eta 10 balioak dituzten hitzak kenduz gero, 0.4 s behar du.
- 8, 9 eta 10 balioak dituzten hitzak kenduz gero, 0.16 s behar du.

Izan ere, 10 balioa duten 236201 hitz daude hiztegian, 9 balioa duten 81107 eta 8 balioa duten 35728. Hori dela eta, balio horiek dituzten hitzak kargatuz gero hiztegia gordetzeko denbora asko handituko da. Hiztegia kargatzeko segundo bat baino gutxiago erabiltzea erabaki dut, beraz, 9 eta 10 balioak dituzten hitzak kenduko ditut.

Ondoren, taularen tamainari dagokionez, 5x5 eta 20x20 arteko tamaina duen taula erabiltzea iruditu zait logikoena. Bi taula horiek sortu eta inprimatzeko behar duten denboraren artean ez dago desberdintasun nabarmenik (5x5: 6 ms eta 20x20: 14 ms). Hori ikusita, 8x13 tamainako taula erabiltzea erabaki dut, 100 karaktere inguru izanik hitz desberdin asko izateko probabilitate ona baitu (5x5ekoak baino askoz handiagoa) eta jokalaria ez zaio handiegia irudituko. Neurri hori erabiliz 8 ms inguru beharko dira taula sortu eta inprimatzeko, beraz, jokalaria erantzat ez da arazorik izango.

Ondorioz, hitzak taulan bilatzeari dagokionez, 8x13 tamainako taula erabiliz hitz bat bilatzeko gehienez 200 μ s inguru behar dituela ikusi dut. Horretarako, hainbat aldiz exekutatu dut aplikazioa, eta taularen posizio ezberdinetan tamaina askotariko hitzak bilatu ditut. Kasu gehienetan 80 eta 160 μ s arteko denborak lortu ditut, beraz, jokalaria ez du nabarituko aplikazioak hitza taulan bilatzeko behar duen denbora.

Gainera, hitzak hiztegian bilatzeko behar den denbora 20 eta 100 μ s artekoa izaten da, eta taulan bilatzeko denbora bezala, jokalaria ez du nabarituko.

Azkenik, sistemak erabiltzailearen aurka jokatzeko behar duen denbora aztertu dut. Denbora hau da guztietatik aldakorrena: proba gehienetan 0.01 ms eta 60 ms arteko emaitzak lortu ditut. Izan ere, taulan ez dagoen hitz bat (0 puntu) edo puntu oso gutxi balio duen hitz bat aukeratzen badu jokalaria, aplikazioak oso azkar aurkituko du hobe den hitz bat (1 ms baino azkarrago). Taulan dagoen eta puntu asko balio dituen hitz bat aurkitzen badu, ordea, aplikazioak hitz gehiago irakurri beharko ditu, eta orokorrean denbora gehiago beharko du. Hala ere, denbora luzeena 60 ms ingurukoa da 8x13 dimentsioko taularekin, beraz, ez da oso garrantzitsua.

Hori guztia dela eta, jokoa orokorrean nahiko azkarra da, nabarmena den denbora bakarra hiztegia kargatzeko behar dena da, irakurtzen duen fitxategiak hitz asko dituelako. Hitz horiek mugatuz gero, jokoa oso azkarra izango da.