

**Maider AZANZA SESÉ**

Donostiako Informatika Fakultateko irakasle eta ikertzailea, UPV/EHU

Software probak: nahi eta nahi ez

1 976ko udara, Olinpiar Jokoak. Uztailaren 18an **Nadia Comaneci barra asimetricoaren ariketa bukatzen ari zen.** Irteera perfektua. Txalo zaparrada entzun zen eta epaileek nota eman zuten. Markagailuan, “1.0”. Zer demontre? Epaileek laster argitu zuten beren kalifikazioa “10.0” zela. Baina sistemak ezin zuen hori erakutsi, diseinatu zutenean ez zutelako pentsatu Olinpiar Jokoetan lor zitekeenik. Historiako lehena izan zen.

Civilization bideojokoan badago antzeko istorio bat. Bertan, hainbat naziotako liderrek agresibitate maila jakin bat zuten, 1 eta 12 artekoa. Demokrazia iristean, maila bat jaisiten zitzaien agresibitate hori liderrei. Gauza da, Gandhi-ren kasuan, bere hasierako maila 1ekoa zela eta bat kenduta, eta tarte 1etik 12rako izanda, 12ra pasatzen zela, “Gandhi nuklear” bihurtuz eta inguruko guztiei arma atomikoekin eraso eginez. Egia da hori legenda dela, nahiz eta jokalaria askok sinetsi, baina ondo erakusten du software batean murretan dauden balioak probatzearen garrantzia.

Aurreko bi gertaerak anekdotak dira, baina sistemak gaizki diseinatzearen ondorioak askoz larriagoak izan daitezke. Mars Climate Orbiter NASAk jaurtitako espazio-zunda robotiko bat zen, Marten klima, atmosfera eta azaleko aldaketak aztertze-ko helburua zuena. 1999ko irailaren 23an, espazio-ontziarekiko komunikazioa galdu egin zen, Marteren orbitan sartutakoan, eta suntsitu egin zen. Bi neurketa-sistemaren arteko nahasketa bati egotzi zioten akatsa: NASaren SI unitatea (sistema metrikoa) eta AEBetako ohiko unitatea, espazio-ontziaren erakitzzaile Lockheed Martinek erabilitakoa. Aldez aurretik egindako probetan ez zen nahasketa hori antzeman.

Softwarea ez da plazaratu baino lehen bakarrik probatu behar. Askotan, bere bizitzan zehar, aldaketak egin behar zaizkio, eta ho-

rien ondoren ondo funtzionatzen jarraitzen duela ziurtatu behar da. Gogoratu, adibidez, pezetak alde batera utzi eta euroak izatera pasatu ginenekoa, 2002. urtekoa. Ordura arte diruarekin lan egiten zuten sistema guztiek zenbaki osoekin egiten zuten lan, pezetek ez baitzuten komarrik. Baina eurora pasatu behar genuenean, denek zenbaki errealekin –hots, komekin– lan egin ahal izateko eboluzio-natu behar izan zuten eta ondo zebiltzala ziurtatu.

Lau adibide besterik ez ditugu aipatu. Orain pentsatu zenbat software erabiltzen duzun zure eguneroko bizitzan: mugikorrean daramatzazun aplikazioetatik zure kotxea kontrolatzen duen softwarea, edo zure garbigailuko programetatik ospitaletako robot kirurgikoetara. Guztiek espero bezala funtziona dezaten proba prozesu zorrotzak pasatu behar izan dituzte. Eta noski, softwarea zenbat eta kritikoagoa izan, orduan eta zorrotzagoa da prozesua, bere esku bizitzak dituen softwarea barne.

Laburbilduz, softwarearen probak funtsezkoak dira gure gizarte teknologikoan. Ezinbestekoak dira, bai garapenean, bai aldaketa bakoitzaren ondoren. Etengabeko hobekuntza eta zorrotasuna behar dira, gure teknologia gero eta konplexuagoa eta kritikoagoa bihurtzen ari den heinean. •



Softwarea aldatzen joan behar da eta ongi funtzionatzen jarraitzen duela ziurtatzen. Ngampol THONGSAI | GETTY IMAGES