Kuidas arvuti juhuslikkust loob, kui arvuti ise juhuslik pole?

1 kuu tagasi - 16.09.2026 Autor: AM

(Sisuturundus)

Arvuti täidab käske täpselt selles järjekorras, nagu programm ette näeb. Seetõttu tekib lihtne küsimus: kust tulevad juhuslikud arvud, mida kasutavad parooligeneraatorid, simulatsioonid või mängud? Vastus sõltub sellest, millist juhuslikkust rakendus vajab ja kust algandmed võetakse.

Mängudes on juhuslikkus päris töövahend

Mängija näeb ekraanil kaarte, sümboleid või mõnda muud tulemust. Tarkvara jaoks tähendab see tavaliselt arvulist väärtust, mille põhjal programm valib järgmise sündmuse. Veebikasiinode puhul on RNG üks tehniline osa suuremast süsteemist, kuhu kuuluvad ka mängureeglid ja serveripoolne loogika.

Eestis tegutsevaid kasiinosaite koondav Netikasiinod keskendub muu hulgas veebikasiinodele ja nende mänguvalikule. Selliste mängude tehnilist poolt selgitades jõuab kiiresti juhuarvugeneraatorini, sest digitaalne slot või kaardimäng vajab iga vooru jaoks uut tulemust. OlyBeti juhuslike numbrite generaatori selgituses kirjeldatakse näiteks, kuidas RNG seob genereeritud arvu sloti sümbolite või kaardimängu tulemusega.

Sellest ei tasu teha järeldust, et RNG kuulub ainult kasiinotarkvara juurde. Sama põhimõtet kasutatakse väga erinevates programmides, kus järgmine väärtus ei tohi olla ette määratud.

Pseudojuhuslik arv algab ühest väärtusest

Kõige tavalisem lahendus on pseudojuhuslike arvude generaator ehk PRNG. Programm alustab lähteväärtusest, mida nimetatakse seemneks ehk seed'iks. Seejärel arvutab algoritm järgmise väärtuse ning kasutab saadud olekut järgmise arvu loomiseks.

Sama algoritm annab sama seemnega sama jada. See omadus on mõnes olukorras väga kasulik. Arendaja saab vea taastamiseks simulatsiooni uuesti käivitada täpselt sama juhuarvude jadaga. Üks lihtsustatud tööjärjekord näeb välja selline:

- Seeme. Programm saab algväärtuse, millest arvutamine algab.
- Algoritm. Matemaatiline reegel muudab sisemise oleku uueks väärtuseks.
- Väljund. Osa saadud väärtusest kasutatakse juhuarvuna.
- Uus olek. Generaator jätkab järgmise arvu arvutamist juba muutunud seisust.

Selline generaator võib toota miljoneid väärtusi väga kiiresti. Hea PRNG annab statistiliselt ühtlase tulemuse ning selle periood on piisavalt pikk, et jada tavakasutuses korduma ei hakkaks.

Programmeerija jaoks tähendab see ka üht olulist praktilist asja. Testis võib seemne teadlikult fikseerida, päris kasutuses tuleb selle valik hoolikamalt läbi mõelda.

Seeme vajab head algmaterjali

Kui programm kasutaks seemnena alati näiteks arvu 12345, oleks kogu järgnev jada korratav. Turvalisust nõudvas süsteemis oleks see halb lähtekoht, sest teine osapool võiks sama algoritmi ja seemne korral saada samad väärtused.

Siin tuleb mängu entroopia. Arvutis tähendab see raskesti ennustatavat sisendit, mida süsteem kogub erinevatest allikatest. Operatsioonisüsteem võib kombineerida näiteks seadmete ajastust, katkestuste infot ja riistvara loodud juhuslikke andmeid.

Hea algmaterjal peab sisaldama piisavalt ennustamatust. Üksnes praegune kellaaeg on nõrk valik, sest võimalikku väärtuste hulka saab oluliselt kitsendada. Mitme sõltumatu allika ühendamine teeb algseisundi taastamise palju keerulisemaks.

Krüptograafilise generaatori puhul kontrollitakse veel üht omadust: varasemate väljundite nägemine ei tohi anda praktilist võimalust järgmise väärtuse arvutamiseks.

Riistvara saab mõõta füüsilist müra

Mõnikord võetakse juhuslikkuse lähteallikaks füüsiline protsess. Selleks kasutatakse riistvaralist juhuarvugeneraatorit ehk TRNG-d. Seade mõõdab nähtust, mille täpset järgmist väärtust pole lihtne ette arvutada.

Allikaks võivad olla näiteks elektrooniline müra või muud füüsilised kõikumised. Mõõdetud signaali töödeldakse enne kasutamist, sest toorandmetes võib esineda kallutatust või seadme enda põhjustatud mustreid.

Praktikas kasutatakse riistvaralist ja pseudojuhuslikku lähenemist sageli koos. Füüsilisest allikast saadud andmed sobivad seemne loomiseks, PRNG suudab seejärel kiiresti suure hulga väärtusi toota. Nii ei pea iga juhuarvu jaoks eraldi füüsilist mõõtmist ootama.

Simulatsioonis võib korratavus olla eelis

Juhuarvud on olulised ka siis, kui midagi ei üritata salajas hoida. Füüsika-, liiklus- ja finantsmudelid kasutavad neid võimalike olukordade läbimängimiseks. Üks simulatsioon võib sisaldada tuhandeid või miljoneid juhuslikke valikuid.

Siin võib fikseeritud seeme olla väga praktiline. Kui kaks arendajat käivitavad sama mudeli sama seemnega, saavad nad kontrollida täpselt sama sündmuste jada. Algoritmi muutmise järel on lihtsam näha, milline osa tulemusest tegelikult muutus.

Näiteks liiklusmudelis võib juhuarv otsustada, millal auto ristmikule jõuab. Teises programmis määrab see prooviks genereeritud andmestiku väärtused. Juhuslikkus aitab mudelil erinevaid olukordi läbi mängida, korratav seeme aitab vea hiljem üles leida.

Krüptograafias on nõuded palju karmimad

Paroolide, krüptovõtmete ja autentimistunnuste juures ei piisa generaatorist, mis lihtsalt näeb statistiliselt juhuslik välja. Tulemus peab olema ka ründaja jaoks praktiliselt ennustamatu.

Seetõttu kasutatakse krüptograafiliselt turvalisi pseudojuhuarvugeneraatoreid. Need saavad seemne süsteemi entroopiaallikatest ning on loodud nii, et väljundite põhjal oleks sisemise oleku taastamine väga raske. Riigi Infosüsteemi Ameti turvalise arenduse suunistes rõhutatakse samuti, et juhuarvugeneraatori seeme peab olema ettearvamatu, sest sama seeme võib anda sama arvujada.

Tavaline rakendus ei pea ise füüsilist müra mõõtma ega uut juhuarvualgoritmi leiutama. Operatsioonisüsteemid pakuvad selleks valmis turvalisi liideseid. Krüptograafilises koodis on enda kirjutatud „lihtne juhuarvugeneraator“ enamasti täiesti tarbetu risk.

Üks tehnoloogia, väga erinevad ülesanded

RNG-d kohtab mängudes, turvatarkvaras, teadusmudelites ja testandmete loomisel. Tallinna Tehnikaülikooli töös online-hasartmängude kohta käsitletakse internetipõhiseid hasartmänge eraldi digitaalse teenusena, mille toimimine sõltub tehnilistest ja regulatiivsetest lahendustest.

Kõigis neis rakendustes ei kasutata sama tüüpi generaatorit. Test võib vajada teadlikult korratavat jada, parooligeneraator aga tugevat entroopiat ja ennustamatust. Seetõttu tasub sõna „juhuslik“ nähes alati vaadata, kuidas algväärtus saadakse ja mille jaoks genereeritud arvu kasutatakse.

- Uudised
- Sisuturundus

Pilt

