

Teme de proiect pentru laboratorul de

Tehnici de Simulare

Grupele 341, 342, 343 și 344

Observatii:

- 1) Toate proiectele vor fi realizate în limbajul **R**, fără excepție!
- 2) Punctajul total asociat proiectului este **40 de puncte(10 puncte pentru interfața Shiny App și 30 de puncte pentru partea algoritmică și documentație)** și va fi detaliat pentru fiecare echipă în parte
- 3) Cerințele de proiect sunt prezentate generic acestea urmând a fi individualizate pentru fiecare echipă în parte, odată ce tema a fost aleasă.
- 4) Fiecare echipa va primi cerințe particulare despre repartițiile și parametrii cu care trebuie să lucreze.
- 5) Termenul de predare al proiectului este **30 aprilie 2021, ora 20:00**

Notă: În continuare, prin **server** vom înțelege o denumire generică dată unui element dintr-un sistem care îndeplinește funcția de a oferi o anumită funcționalitate către un **client**, acesta din urmă fiind, de cele mai multe ori, o persoană. Rolul de **server** poate fi jucat de o componentă fizică, de un program informatic sau de o persoană având un rol specific la locul de muncă.

Cerințe de proiect la Tehnici de Simulare

Punctajul asociat proiectului va fi acordat după următorul barem:

- 1) Folosirea interfeței **Shiny App 10 p**
- 2) **Documentația (element eliminativ, proiectele fără documentație nu vor fi evaluate!) 15 p**
 - Reprezentantul echipei va trimite la **simona.cojocsa@fmi.unibuc.ro** proiectul realizat de echipă care este format din 2 fișiere: un document **.pdf** care să conțină documentația și un (singur!) fișier **.R** care să conțină codul comentat*
 - Documentația trebuie să conțină:
 - ✓ descrierea problemei
 - ✓ datele problemei(cele furnizate de mine împreună cu eventualele ipoteze suplimentare pe care le formulați voi)
 - ✓ rezolvarea eventualelor probleme teoretice date
 - ✓ codul comentat
 - ✓ rezultatele obținute în urma simulării și interpretarea lor(cu grafice relevante)
 - ✓ comentarii și concluzii
 - Studenții care identifică și rezolvă corect *cerințe suplimentare* ale problemei de simulare față de cele obligatorii pot obține un **bonus de 5p sau 10p**, în

funcție de complexitatea și oportunitatea cerinței suplimentare formulate.

3) Implementarea corectă a temei de simulare(respectarea specificațiilor proiectului, folosirea eficientă a limbajului R) **10 p**

4) Folosirea corectă a metodelor de simulare **5 p**

OBS: Puteți folosi doar funcția **runif** din R pentru generarea unei valori dintr-o repartiție uniformă, pentru toate celelalte repartiții cu care lucrați trebuie să scrieți propria funcție în R (acolo unde nu este în mod expres cerută folosirea unei anumite metode de simulare alegerea vă aparține).

***Necomentarea** codului atrage după sine pierderea a **5 p**. Comentariile trebuie să permită o parcurgere facilă a codului, ajutând la delimitarea diferitelor module din proiect, precum și explicarea utilității diferitelor variabile folosite și funcționalitatea anumitor bucăți de cod.

1. Sistem de tip coadă cu două servere legate în serie

Considerăm un sistem cu două servere secvențiale(*în tandem*). Funcționarea acestuia este următoarea:

- La sosirea unui client, dacă **serverul 1** este liber va servi imediat clientul, altfel acesta din urmă va intra în coada de așteptare a **serverului 1**.
- După executarea sarcinii necesare de către **serverul 1** procesul se repetă și în raport cu **serverul 2** (**Atenție**: cele două servere trebuie folosite doar în această ordine!).
- După finalizarea execuției de către **serverul 2** clientul părăsește sistemul.

Convenții:

- Sosirea clienților se face conform unui **proces Poisson neomogen** cu **funcția de intensitate $\lambda(t)$** .
- *Timpii de servire* asociați celor două servere sunt considerate variabile aleatoare având funcțiile de repartiție **G_1** și respectiv **G_2** .

▪ **Obiectivele simulării:**

- 1) Determinarea **timpului minim, maxim și mediu** petrecut de un client în sistem(cu detalieri pentru fiecare server în parte)
- 2) Determinarea **numărului mediu** de clienți serviți într-o zi/perioadă de timp prestabilită
- 3) Determinarea **primului moment de timp** la care este pierdut un client
- 4) Determinarea **numărului mediu de clienți pierduți** din cauza timpului de așteptare prea mare(cu detalieri pentru fiecare server în parte)
- 5) Determinarea **câștigului mediu suplimentar** care ar fi obținut dacă s-ar mări programul de lucru cu o oră(cu detalieri pentru începerea programului mai devreme și respectiv pentru prelungirea cu o oră a programului curent) și respectiv dacă s-ar mări dimensiunea maximă a cozii de așteptare.

2. Sistem de tip coadă cu două servere legate în paralel

Considerăm un sistem cu două servere ce îndeplinesc aceeași funcție, în mod similar. Funcționarea acestuia este următoarea:

- La sosirea unui client, dacă **serverul 1** este liber va servi imediat clientul, altfel acesta se adresează **serverului 2** și în cazul în care și acesta este ocupat clientul intră într-o coadă de așteptare comună celor 2 servere.
- După finalizarea execuției de către **serverul 1** sau **2**, după caz, clientul părăsește sistemul și clientul cu timpul de așteptare cel mai mare din coadă urmează să fie deservit de serverul respectiv.

Convenții:

- Sosirea clienților se face conform unui **proces Poisson neomogen** cu **funcția de intensitate $\lambda(t)$** .
- *Timpul de servire* asociați celor două servere sunt considerate variabile aleatoare având funcțiile de repartiție **G_1** și respectiv **G_2** .
- **Obiectivele simulării:**
 - 1) Determinarea **timpului minim, maxim și mediu** petrecut de un client în sistem(cu detalieri pentru fiecare server în parte)
 - 2) Determinarea **numărului mediu de clienți serviți** de fiecare server, precum și al **numărului mediu total de clienți** serviți de sistem într-o anumită perioadă de timp.
 - 3) Determinarea **primului moment de timp** la care este pierdut un client
 - 4) Determinarea **numărului mediu de clienți pierduți** din cauza timpului de așteptare prea mare(cu detalieri pentru fiecare server în parte)
 - 5) Determinarea **câștigului mediu suplimentar** care ar fi obținut dacă s-ar mări programul de lucru cu o oră(cu detalieri pentru începerea programului mai devreme și respectiv pentru prelungirea cu o oră a programului curent) și respectiv dacă s-ar mări dimensiunea maximă a cozii de așteptare.

3. Un model de inventar

Considerăm un magazin care ține pe stoc un anumit produs care se vinde cu *prețul r per unitate de produs*. Pentru a face față cererii pentru produsul respectiv, gestionarul trebuie să păstreze o anumită cantitate disponibilă în stoc, iar atunci când aceasta se diminuează trebuie să facă o nouă comandă către distribuitor.

Convenții:

- Solicitățile clienților pentru produsul respectiv apar conform unui **proces Poisson neomogen** cu **funcția de intensitate $\lambda(t)$** .
Cantitatea solicitată de fiecare client este o variabilă aleatoare cu funcția de repartiție **G**
- Gestionarul aplică o politică de comandă către furnizor de tipul **(s, S)** , ceea ce înseamnă că atunci când stocul scade sub cantitatea **s** și nu are nicio comandă curentă de onorat va plasa o comandă către distribuitor astfel încât să ajungă stocul la cantitatea **S** , unde **$s < S$** .
- Costul asociat cu **y** unitați din produsul respectiv este dat de funcția **$c(y)$** , timpul până la livrare este notat cu **L** , iar plata este făcută la livrare.
- Magazinul are și un cost unitar **h** pe unitatea de timp pentru păstrarea produsului în inventar
- Lucrăm cu ipoteza că atunci când un client face o comandă mai mare decât cantitatea disponibilă în stoc, magazinul îi livrează cantitatea disponibilă, restul comenzii fiind anulată.
- **Obiectivele simulării:**
 - 1) Determinarea **profitului mediu** până la un moment de timp fixat **T** .
 - 2) Determinarea **numărului mediu** de comenzi primite și care nu au putut fi onorate, per unitate de timp(de unde putem deduce **pierdere medie** înregistrată).
 - 3) Determinarea **numărului mediu de clienți pierduți** din cauza stocului indisponibil până la un moment de timp fixat **T**
 - 4) Determinarea **timpului minim mediu** până la care se vând un număr prestabilit de produse **a** .
 - 5) Determinarea **valorii minime a** a stocului la momentul inițial astfel încât până la momentul **T** nicio comandă să nu fie pierdută sau livrată parțial(folositi cel puțin 10^6 rulări a simulării).

4. Un model de asigurare de risc

Considerăm o firmă de asigurări ce vinde un singur tip de poliță de asigurare către clienții săi(ex. asigurare împotriva îmbolnavirii cu Covid-19). Dorim să studiem, prin intermediul simulării, cât de eficientă ar fi vânzarea acestui produs.

Convenții:

- Pornim cu un număr inițial de n_0 clienți și un capital a_0 .
- Clienții fac cereri de despăgubire conform unui **proces Poisson omogen de rată λ**
- Valoarea fiecărei despăgubiri solicitate este o variabilă aleatoare cu funcția de repartiție **F**
- Potențialii noi clienți ai firmei semnează contracte de asigurare conform unui **proces Poisson omogen de rată v**
- Actualii clienți ai firmei rămân fideli firmei pentru un timp aleator ce corespunde repartiției **exponențiale** de parametru μ
- Toți clienții firmei plătesc o sumă fixă **c** per unitate de timp.
(ex. 200 lei/lună)
- **Obiectivele simulării:**
 - 1) Estimarea probabilității ca firma să **nu** fie **ruinată** până la un moment de timp **T** fixat.(i.e. capitalul firmei până la momentul T este mereu pozitiv!)
 - 2) Estimarea unui **capital minim** necesar pentru ca probabilitatea estimată de ruină până la momentul T să fie mai mică de 30%.
 - 3) Determinarea **timpului mediu** după care un client pleacă la altă firmă de asigurare
 - 4) Estimarea **probabilității** ca până la un moment de timp fixat T mai mulți clienți să părăsească firma de asigurări decât să fie semnate contracte cu noi clienți
 - 5) Estimarea **probabilității** ca toți clienții noi dintr-un interval prestabilit de timp($[T-t, T]$) să solicite despăgubiri în următoarele **s** unități de timp.

