

# analyse de performance et simulation - M1 II et ISIFAR

## Examen du 19 mai 2009

Université Paris Diderot

Tous les documents sont autorisés  
Durée : 2h30

### Exercice 1 – Génération aléatoire

On dispose d'une fonction `double u()` qui renvoie un réel de double précision aléatoire de loi uniforme sur  $(0;1)$ . Programmez les fonctions qui génèrent les variables aléatoires suivantes :

1. X une pièce courbée, qui tombe sur pile (0) avec probabilité 0,36 ; sur face (1) avec probabilité 0,6 ; et sur le bord (2) avec probabilité 0,04
2. Y le temps d'attente d'un météorite - - en années (loi exponentielle avec  $\lambda = 0.04$ )
3. Z de loi de Cauchy, avec la fonction de distribution  $F(x) = \frac{1}{2} + \frac{1}{\pi} \arctan(x)$

### Exercice 2 – File d'attente

On considère une file d'attente M/M/3.

1. Expliquez brièvement le sens de cette notation.
2. Le temps de service moyen par un serveur est  $\bar{S} = 6s$ , le nombre de requêtes moyen est  $\lambda$  par seconde. Trouvez quelles valeurs de  $\lambda$  la file peut supporter en restant stable. Quel théorème utilisez-vous ?
3. Formulez la loi de Little pour cette file, en expliquant tous les paramètres.

### Exercice 3 – Analyse de bottleneck

On considère un réseau de files d'attentes qui représente un hôpital. On suppose qu'il y a toujours  $n$  patients dans l'hôpital. Chaque malade est d'abord reçu par l'accueil avec le temps de réflexion  $Z=10$  minutes, ensuite il visite les cabinets A (en moyenne  $V_A = 0.5$  fois), B (en moyenne  $V_B = 0.25$  fois), C (en moyenne  $V_C = 3$  fois). Les temps moyens de service dans chaque cabinet sont 10, 15 et 2 minutes respectivement.

1. Estimez (en utilisant les lois opérationnelles) comment le nombre de patients  $\lambda$  traités par minute dépend de  $n$  et dessinez un graphe.
2. Mêmes questions pour le temps moyen de réponse.
3. Quel cabinet est le maillon faible (bottleneck) qui limite la performance de l'hôpital ?

### Exercice 4 – Collapse

Les clients de type 1 et 2 arrivent à l'accueil de la banque ( respectivement  $\lambda_1 = 5$  et  $\lambda_2$  clients par minute). L'accueil traite  $c_a = 10$  clients par minute et les renvoie aux guichets G1 (pour le type 1) et G2 (pour le type 2). La capacité de chaque guichet est  $c_{G1} = c_{G2} = 5$  clients par minute.

1. En utilisant la modélisation fluide représentez le système comme un réseau de flux. Faites un dessin et indiquez clairement les sources, les destinations et les capacités des "tuyaux".
2. Etudiez le fonctionnement du système pour  $\lambda_2 = 1$ , pour  $\lambda_2 = 5$ , pour  $\lambda_2 = 10$ ,  $\lambda_2 = 100$  et trouvez le débit D (nombre de clients traités par minute) .
3. Essayez de trouver une formule pour D en fonction de  $\lambda_2$ .
4. Dessinez le graphe de dépendance de D en fonction de  $\lambda_2$ .
5. Décrivez le phénomène observé. Comment s'appelle t'il ?
6. Comment peut-on éviter ce phénomène grâce à un accueil intelligent ? Proposez les règles optimales pour l'accueil et dessinez le graphe de débit résultant.

### Exercice 5 – ns2

Lisez le programme ns2 suivant. Sur le recto de la page du programme commentez le code. Sur le verso dessinez le réseau simulé par ce code. Insérez le programme dans votre copie.

```

set ns [new Simulator]

$ns color 1 Blue
$ns color 2 Red

set nf [open out.nam w]
$ns namtrace-all $nf

proc finish {} {
    global ns nf
    $ns flush-trace
    close $nf
    exec nam out.nam &
    exit 0
}

set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]

$ns duplex-link $n0 $n2 1Mb 10ms DropTail
$ns duplex-link $n1 $n2 1Mb 10ms DropTail
$ns duplex-link $n3 $n2 1Mb 10ms SFQ

$ns duplex-link-op $n0 $n2 orient right-down
$ns duplex-link-op $n1 $n2 orient right-up
$ns duplex-link-op $n2 $n3 orient right

#Monitor the queue for the link between node 2 and node 3
$ns duplex-link-op $n2 $n3 queuePos 0.5

set udp0 [new Agent/UDP]
$udp0 set class_ 1
$ns attach-agent $n0 $udp0

set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0

set udp1 [new Agent/UDP]
$udp1 set class_ 2
$ns attach-agent $n1 $udp1

set cbr1 [new Application/Traffic/CBR]
$cbr1 set packetSize_ 500
$cbr1 set interval_ 0.005
$cbr1 attach-agent $udp1

set null0 [new Agent/Null]
$ns attach-agent $n3 $null0

$ns connect $udp0 $null0
$ns connect $udp1 $null0

$ns at 0.5 "$cbr0 start"
$ns at 1.0 "$cbr1 start"
$ns at 4.0 "$cbr1 stop"
$ns at 4.5 "$cbr0 stop"
$ns at 5.0 "finish"

$ns run

```