

BÁO CÁO THỰC HÀNH

Môn học: Đánh giá hiệu năng hệ thống mạng máy tính

Buổi báo cáo: Lab 02

Tên chủ đề: Queuing

GVHD: Đặng Lê Bảo Chương

Ngày thực hiện: 15/10/2025

THÔNG TIN CHUNG:

Lớp: NT531.Q11.2

STT	Họ và tên	MSSV	Email
1	Nguyễn Lê Như Thuận	23521551	23521551@gm.uit.edu.vn

1. ĐÁNH GIÁ KHÁC:

Nội dung	Kết quả
Tổng thời gian thực hiện bài thực hành trung bình	3 tiếng
Link Video thực hiện (nếu có)	
Ý kiến (nếu có) + Khó khăn + Đề xuất ...	+ Trong bài làm này ở một số câu em có dùng nhiều hàm của EXCEL để thể hiện tính đa dạng và cũng vì không biết cách nào là linh hoạt nhất ➔ Bài LAB này giúp em tìm hiểu thêm về nhiều Hàm trong EXCEL <3
Điểm tự đánh giá	10/10

Phần bên dưới của báo cáo này là báo cáo chi tiết của nhóm thực hiện.

Mục lục

1. Tính $\lambda = A/tm:$	3
2. Tính $q:$ IF $x < n:$ $q=0$ /Else $q=x-n$	3
3. Tính $\lambda x:$	3
4. Tính $\mu x:$	4
5. Tính $Tx:$ (Non-normalized term).....	4
6. Tính $Px = Tx / (\text{sum of } Tx)$ sử dụng công thức bên dưới	5
7. Tính $\lambda(x)*P(x)$	6
8. Tính λT (The total arrival rate)	7
9. $P\{\text{immediate service}\}$	7
10. $P\{\text{delayed service}\}$	8
11. $P\{\text{loss}\}$	9
12. Tính $E\{q\}:$ độ dài trung bình của hàng đợi	10
13. Tính luôn các ô chưa có kết quả	12
14. Thay đổi các thông số bao gồm A, K, N để ra các trường hợp sau:.....	12
a) Hệ rảnh và cường độ thấp:.....	13
b) Hệ chưa bị quá tải nhưng cường độ cao:	14
15. Với thông số mặc định trong file excel thay $A = 3$, hãy giải thích vì sao khi $N = n = 5$ thì hệ thống đợi có $P\{\text{immediate service}\} = 1$ và khi băng đầu tăng $N = 6,7,8,9,..., K$ thì $P\{\text{immediate service}\}$ lại tăng nhanh.....	15
YÊU CẦU CHUNG	17

BÁO CÁO CHI TIẾT

1. Tính $\lambda = A/tm$:

- Thực hiện theo công thức: $\lambda = A/tm$

	A	B	C	D
1	Hàng đợi M/M/n/K/N			
2	Khởi tạo:			
3	A	4		
4	n	5		
5	tm (arrival time)	3		
6	K	15		
7	N (No. of customers)	18		
8	λ	1.33		
a				sum of Tx

2. Tính q: IF $x < n$: $q = 0$ Else $q = x - n$

- Thực hiện theo công thức của đề bằng cách sử dụng hàm **IF(biểu thức, [phép tính 1], [phép tính 2])**

Với: Nếu biểu thức đúng \rightarrow Phép tính 1, ngược lại \rightarrow Phép tính 2

	A	B	C	D	E	F	G	H
1	Hàng đợi M/M/n/K/N							
2	Khởi tạo:				Output:			
3	A	4			P{im ser.}			P{x=K}
4	n	5			P{del.}			P{n-1 < x < K}
5	tm (arrival time)	3			P{loss}			P{x < n}
6	K	15			E{q all state}			E{q queue exist}
7	N (No. of customers)	18			E{x}			
8	λ	1.33			sum of Tx			
9								λ_T
10	x	q	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	$(x-n) * P(x)$
11	0	0						
12	1	0						
13	2	0						
14	3	0						
15	4	0						
16	5	0						

3. Tính λ_x :

If $N=0$ (i.e number of sources is infinite \rightarrow M/M/n/K model) then $\lambda_x = \lambda T = \lambda = A/t_m$

Else if $x \leq K$ then $\lambda_x = \lambda * (N-x)$

Else $\lambda_x = 0$ (these states do not exist in the state transition diagram)

- Thực hiện theo công thức của đề và dùng hàm IF trong EXCEL

\Rightarrow Lồng 2 IF vào nhau: $=IF($B$7=0,$B$8,IF(A11<=$B$6, B8*(B7-A11), 0))$

+ Với IF ($N=0$, λ_x , [IF 2])

- [IF 2]: IF ($x \leq K$, $\lambda * (N-x)$, 0)

	A	B	C	D	E	F	G	H
1	Hàng đợi M/M/n/K/N							
2	Khởi tạo:				Output:			
3	A	4			P{im ser.}		P{x=K}	
4	n	5			P{del.}		P{n-1 < x < K}	
5	tm (arrival time)	3			P{loss}		P{x < n}	
6	K	15						
7	N (No. of customers)	18			E{q all state}		E{q queue exist}	
8	λ	1.33			E{x}			
9				sum of Tx		λ_T		
10	x	q	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	$(x-n) * P(x)$
11	0	0	24					
12	1	0	22.6667					
13	2	0	21.3333					
14	3	0	20					
15	4	0	18.6667					
16	5	0	17.3333					

4. Tính μ_x :

If $x > K$ then $\mu_x = 0$ (these states do not exist in the state transition diagram)

Else if $x > n$ then $\mu_x = n\mu = n/t_m$

Else $\mu_x = x\mu = x/t_m$

- Thực hiện theo công thức của đề và dùng hàm IF trong EXCEL
- ➔ Lồng 2 IF vào nhau: =IF(A12>\$B\$6,0, IF(A12>\$B\$4, \$B\$4/\$B\$5,A12/\$B\$5))
 - + Với IF ($x > K, 0$, [IF 2])
 - [IF 2]: IF ($x < N, n/t_m, x/t_m$)

	A	B	C	D	E	F	G	H
1	Hàng đợi M/M/n/K/N							
2	Khởi tạo:				Output:			
3	A	4			P{im ser.}		P{x=K}	
4	n	5			P{del.}		P{n-1 < x < K}	
5	tm (arrival time)	3			P{loss}		P{x < n}	
6	K	15						
7	N (No. of customers)	18			E{q all state}		E{q queue exist}	
8	λ	1.33			E{x}			
9				sum of Tx		λ_T		
10	x	q	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	$(x-n) * P(x)$
11	0	0	24	0				
12	1	0	22.6667	0.3333333				
13	2	0	21.3333	0.6666667				
14	3	0	20	1				
15	4	0	18.6667	1.3333333				
16	5	0	17.3333	1.6666667				
17	6	1	16	1.6666667				

5. Tính T_x : (Non-normalized term)

If $x > K$ then $T_x = 0$ (these states do not exist in the state transition diagram)

Else if $\mu_x \neq 0$ then $T_x = T_{x-1} \cdot (\lambda_{x-1}/\mu_x)$

Else $T_x = 0$

Note:

- Take a large number C of values of T_x , i.e larger than K.
- In this way, we have:
- Initially, $T_0 = 1$.

- Theo đề: Khởi tạo ban đầu T0=1
➔ Các T1 trở về Tn sẽ được tính bằng: =IF(A12>\$B\$6,0,IF(D12=0,0,E11*(C11/D12)))
+ Với IF ($x > K, 0, [IF\ 2]$)
 - [IF 2]: IF ($\mu_x = 0, 0, \lambda x - 1 * (\lambda x - 1) / \mu_x$) //câu này em đảo ngược điều kiện lại

A		B	C	D	E	F	G	H
1	Hàng đợi M/M/n/K/N							
2	Khởi tạo:				Output:			
3	A	4			P{im ser.}		P{x=K}	
4	n	5			P{del.}		P{n-1 < x < K}	
5	tm (arrival time)	3			P{loss}		P{x < n}	
6	K	15						
7	N (No. of customers)	18			E{q all state}		E{q queue exist}	
8	λ	1.33			E{x}			
9					sum of Tx		λ_T	
10	x	q	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	$(x-n) * P(x)$
11	0	0	24	0	1			
12	1	0	22.6667	0.3333333	72			
13	2	0	21.3333	0.6666667	2448			
14	3	0	20	1	52224			
15	4	0	18.6667	1.3333333	783360			
16	5	0	17.3333	1.6666667	8773632			
17	6	1	16	1.6666667	91245772.8			
18	7	2	14.6667	1.6666667	875959419			
19	8	3	13.3333	1.6666667	7708442886			
20	9	4	12	1.6666667	6.1668E+10			

6. Tính $P_x = T_x / (\text{sum of } T_x)$ sử dụng công thức bên dưới

$$P_x = \frac{T_x}{\sum_{i=1}^C T_i}$$

Proof:

$$P_x = \frac{\lambda_0 \lambda_1 \dots \lambda_{x-1}}{\mu_1 \mu_2 \dots \mu_x} \cdot P_0 \text{ (have already shown in Queueing Systems)}$$

And we also have:

$$P_x = \frac{T_x}{\sum_{i=1}^c T_i} = \frac{\frac{\lambda_0 \lambda_1 \dots \lambda_{x-1}}{\mu_1 \mu_2 \dots \mu_x} \cdot T_0}{\sum_{i=1}^c \frac{\lambda_0 \lambda_1 \dots \lambda_{i-1}}{\mu_1 \mu_2 \dots \mu_i} \cdot T_0} = \frac{\frac{\lambda_0 \lambda_1 \dots \lambda_{x-1}}{\mu_1 \mu_2 \dots \mu_x} \cdot P_0}{\sum_{i=1}^c \frac{\lambda_0 \lambda_1 \dots \lambda_{i-1}}{\mu_1 \mu_2 \dots \mu_i} \cdot P_0} = \frac{P_x}{\sum_{i=1}^c P_i} = \frac{P_x}{1} = P_x$$

- Đầu tiên cần tính: $\sum_{i=1}^c Ti$

	A	B	E	F	G	H
1	Hàng đợi M/M/n/K/N		SUM(number1, [number2], ...)			
2	Khởi tạo:			Output:		
3	A	4		P{im ser.}		P{x=K}
4	n	5		P{del.}		P{n-1<x<K}
5	tm (arrival time)	3		P{loss}		P{x<n}
6	K	15				
7	N (No. of customers)	18		E{q all state}		E{q queue e}
8	λ	1.33		E{x}		
9			sum of Tx	= SUM(E12:E111)	λ_T	
10	x	q	λ_x	μ_x	T_x	P_x
11	0	0	24	0	1	7.2522E-16
12	1	0	22.6667	0.3333333	72	5.2216E-14
13	2	0	21.3333	0.6666667	2448	1.7753E-12
14	3	0	20	1	52224	3.7874E-11
15	4	0	18.6667	1.3333333	783360	5.6811E-10
16	5	0	17.3333	1.6666667	8773632	6.3628E-09
17	6	1	16	1.6666667	91245772.8	6.6173E-08
18	7	2	14.6667	1.6666667	875959418.9	6.3526E-07

- Ở mỗi P_x tương ứng: sẽ được tính bằng: $T_x / (\text{sum of } T_x)$

	A	B	C	D	E	F	G	H
1	Hàng đợi M/M/n/K/N							
2	Khởi tạo:				Output:			
3	A	4			P{im ser.}			P{x=K}
4	n	5			P{del.}			P{n-1<x<K}
5	tm (arrival time)	3			P{loss}			P{x<n}
6	K	15						
7	N (No. of customers)	18			E{q all state}			E{q queue e}
8	λ	1.33			E{x}			
9				sum of Tx	1.37889E+15	λ_T		
10	x	q	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	$(x-n) * P(x)$
11	0	0	24	0	1	7.2522E-16		
12	1	0	22.6667	0.3333333	72	5.2216E-14		
13	2	0	21.3333	0.6666667	2448	1.7753E-12		
14	3	0	20	1	52224	3.7874E-11		
15	4	0	18.6667	1.3333333	783360	5.6811E-10		
16	5	0	17.3333	1.6666667	8773632	6.3628E-09		
17	6	1	16	1.6666667	91245772.8	6.6173E-08		
18	7	2	14.6667	1.6666667	875959418.9	6.3526E-07		
19	8	3	13.3333	1.6666667	7708442886	5.5903E-06		
20	9	4	12	1.6666667	61667543089	4.4723E-05		
21	10	5	10.6667	1.6666667	4.44006E+11	0.000322		

7. Tính $\lambda(x)*P(x)$

- Lấy 2 ô tương ứng của 2 cột nhân với nhau

A	B	C	D	E	F	G	H
	5			P{del.}		P{n-1<	
arrival time)	3			P{loss}		P{x<n}	
	15						
of customers)	18			E{q all state}		E{q qu	
	1.33			E{x}			
			sum of Tx	1.37889E+15	λ_T		
x	q	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	$(x-n)^*$
0	0	24	0		1	7.2522E-16	1.74053E-14
1	0	22.6667	0.3333333		72	5.2216E-14	1.18356E-12
2	0	21.3333	0.6666667		2448	1.7753E-12	3.78739E-11
3	0	20	1		52224	3.7874E-11	7.57478E-10
4	0	18.6667	1.3333333		783360	5.6811E-10	1.06047E-08
5	0	17.3333	1.6666667		8773632	6.3628E-09	1.10289E-07
6	1	16	1.6666667		91245772.8	6.6173E-08	1.05877E-06
7	2	14.6667	1.6666667		875959418.9	6.3526E-07	9.3172E-06

8. Tính λ_T (The total arrival rate)

$$\lambda_T = \sum_{x=0}^K \lambda_x * P_x$$

- Theo công thức: sẽ cộng từ ($\lambda_x * P_x$) với x từ 0 → K
- ➔ Sử dụng SUMIF thêm điều kiện: SUMIF(A11:A111,"<="&B6,G11:G111)

A	B	C	D	E	F	G	H	I
1	Hàng đợi M/M/n/K/N							
2	Khởi tạo:			Output:				
3	A	4		P{im ser.}			P{x=K}	
4	n	5		P{del.}			P{n-1<x<K}	
5	tm (arrival time)	3		P{loss}			P{x<n}	
6	K	15						
7	N (No. of customers)	18		E{q all state}			E{q queue exist}	
8	λ	1.33		E{x}				
9			sum of Tx	1.37889E+15	λ_T	4.502875134		
10	x	0	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	$(x-n)^* P(x)$
		0	24	0		1	7.2522E-16	1.74053E-14

9. $P\{\text{immediate service}\}$

$$P\{\text{immediate service}\} = \sum_{x=0}^{n-1} \frac{\lambda_x * P_x}{\lambda_T}$$

- Theo công thức của đề + dùng SUMIF, với điều kiện $0 \leq x \leq n-1$ ($\Leftrightarrow 0 \leq x \leq n$)
- ➔ $P = \sum_{k=0}^{n-1} \frac{\lambda(x)*P(x)}{\lambda_T} = \left(\sum_{k=0}^{n-1} \frac{\lambda(x)*P(x)}{1} \right) / \lambda_T \quad (\text{vì } \frac{x}{a} + \frac{y}{a} = \frac{x+y}{a})$



F3					=SUMIF(A11:A111,"<"&B4,G11:G111)/G9			
A	B	C	D	E	F	G	H	I
1 Hàng đợi M/M/n/K/N								
2 Khởi tạo:				Output:				
3 A	4			P{im ser.}	2.53199E-09		P{x=K}	
4 n	5			P{del.}			P{n-1<x<K}	
5 tm (arrival time)	3			P{loss}			P{x<n}	
6 K	15			E{q all state}			E{q queue exist}	
7 N (No. of customers)	18			E{x}				
8 λ	1.33			sum of Tx	1.37889E+15	λ_T	4.502875134	
9 x	0	q	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	$(x-n) * P(x)$
10			24	0	1	7.2522E-16	1.74053E-14	
11								

❖ Hoặc có thể làm: =SUM(\$G\$11:OFFSET(\$G\$11,\$B\$4-1,0))/G9 (thày gợi ý trong lớp, linh hoạt hơn)

Giải thích: Cộng từ [P_1 tới {P_x (x≤ n-1)}]/G9

với hàm OFFSET có cấu trúc sau: **OFFSET(reference, rows, cols, [height], [width])**

- Reference: bắt đầu
- Rows: số lượng ô dịch xuông (ở bài này là dịch xuông n-1= 3 hàng)
- Cols: dịch ngang trái phải (ở bài này dịch 0)
- [height] và [width]: chiều cao độ rộng ô tham chiếu trả về (câu này không cần điền 2 tham số này)

F3					=SUM(\$G\$11:OFFSET(\$G\$11,\$B\$4-1,0))/G9			
A	B	C	D	E	F	G	H	I
1 Hàng đợi M/M/n/K/N								
2 Khởi tạo:				Output:				
3 A	4			P{im ser.}	2.53199E-09		P{x=K}	
4 n	5			P{del.}			P{n-1<x<K}	
5 tm (arrival time)	3			P{loss}			P{x<n}	
6 K	15			E{q all state}			E{q queue exist}	
7 N (No. of customers)	18			E{x}				
8 λ	1.33			sum of Tx	1.37889E+15	λ_T	4.502875134	
9 x	0	q	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	
10			24	0	1	7.2522E-16	1.74053E-14	
11	1	0	22.6667	0.3333333	72	5.22158E-14	1.18356E-12	
12	2	0	21.3333	0.66666667	2448	1.77534E-12	3.78720E-11	
13								

10. P{delayed service}

$$P\{\text{delayed service}\} = \sum_{x=n}^{K-1} \frac{\lambda_x * P_x}{\lambda_T}$$

- ❖ Khá giống P{immediate service}, nhưng x được bắt đầu từ n và kết thúc K-1 ($n \leq x < K$):
- ❖ Sử dụng SUMIFS: với vùng dữ liệu cần tính tổng là từ G11:G111 kèm 2 điều kiện là chỉ lấy các hàng có $x > n-1$ và $x < K$

Clipboard		Styles	Add-ins				
F4		=SUMIFS(G11:G111, A11:A111, ">"&B4, A11:A111, "<"&B6)/G9					
	A	B	C	D	E	F	G
1	Hàng đợi M/M/n/K/N						
2	Khởi tạo:				Output:		
3	A	4			P{im ser.}	2.53199E-09	
4	n	5			P{del.}	0.370133858	
5	tm (arrival time)	3			P{loss}		
6	K	15					
7	N (No. of custumers)	18			E{q all state}		
8	λ	1.33			E{x}		
9				sum of Tx	1.37889E+15	λ_T	4.502875134
10							

- ❖ Sử dụng SUM + OFFSET (cách thầy gợi ý trên lớp):

 $=\text{SUM(OFFSET($G$11,$B$4,0):OFFSET($G$11,$B$6-1,0))}/G9$
 - Nhìn chung thì 2 cụm OFFSET này để xác định được ô bắt đầu ($x=n$) và ô kết thúc ($x=K-1$) cho việc cộng lai bằng hàm SUM

F4	X	✓	fx	=SUM(OFFSET(\$G\$11,\$B\$4,0):OFFSET(\$G\$11,\$B\$6-1,0))/G9			Add-ins
A	B	C	D	E	F	G	
Hàng đợi M/M/n/K/N							
Khởi tạo:				Output:			
A	4			P{im ser.}	2.53199E-09		
n	5			P{del.}	0.370133882		
tm (arrival time)	3			P{loss}			
K	15						
N (No. of customers)	18			E[q all state]			

11. P{loss}

$$P\{\text{loss}\} = \frac{\lambda_K * P_K}{\lambda_T} \quad (P_K = P_{(x=k)})$$

- Tính P(x=k): **INDEX(F11:F111, MATCH(B6, A11:A111, 0))**
 - Sử dụng INDEX kết hợp MATCH:
 - + Hàm *MATCH(giá trị tìm kiếm, mảng tìm kiếm, [kiểu khớp])*: trả về số thứ tự của ô kiếm được – được đếm từ vùng chỉ định → bài này trả về 16, vì vùng chỉ định là từ 0→100 và giá trị tìm kiếm là 15
 - + Hàm *INDEX(array, row_num, [column_num])*: trả về giá trị của ô (trong khoảng array) ứng với hàng vừa tìm ở MATCH, bài này không xét column

I3	X	✓	fx	=INDEX(F11:F111, MATCH(B6, A11:A111, 0))				
A	B	C	D	E	F	G	H	I
1 Hàng đợi M/M/n/K/N								
2 Khởi tạo:				Output:				
3 A	4			P{im ser.}	2.53199E-09		P{x=K}	0.709052117
4 n	5			P{del.}			P{n-1<x<K}	
5 tm (arrival time)	3			P{loss}			P{x<n}	
6 K	15							
7 N (No. of customers)	18			E{q all state}			E{q queue exist}	
8 λ	1.33			E{x}				
9				sum of Tx	1.37889E+15	λ_T	4.502875134	
10 x	q	λ_x	μ_x	T_x		P_x	$\lambda_x * P_x$	$(x-n) * P(x)$
11 0	0	24	0	1	7.2522E-16		1.74053E-14	
12 1	0	22.6667	0.33333333	72	5.22158E-14		1.18356E-12	
13 2	0	21.3333	0.66666667	2448	1.77534E-12		3.78739E-11	
14 3	0	20	1	52224	3.8739E-11		7.57478E-10	

- Tính $\lambda_x \{x=K\}$ (em thêm này vô bảng chung luôn): tính giống tính $P\{x=k\}$

I2											
	C	D	E	F	G	H	I	J			
1											
2			Output:			$\lambda_x \{x=K\}$	4				
3			$P\{\text{im ser.}\}$	2.53199E-09		$P\{x=K\}$	0.709052117				
4			$P\{\text{del.}\}$	0.370133882		$P\{n-1 < x < K\}$					
5			$P\{\text{loss}\}$	0.629866115		$P\{x < n\}$					

→ Tính $P\{\text{loss}\}$ theo công thức:

I2													
	C	D	E	F	G	H	I	J					
1													
2			Output:			$\lambda_x \{x=K\}$	4						
3			$P\{\text{im ser.}\}$	2.53199E-09		$P\{x=K\}$	0.709052117						
4			$P\{\text{del.}\}$	0.370133882		$P\{n-1 < x < K\}$							
5			$P\{\text{loss}\}$	0.629866115		$P\{x < n\}$							
6													
7			$E\{q \text{all state}\}$				$E\{q \text{queue exist}\}$						
8			$E\{x\}$										
9			sum of Tx	1.37889E+15		λ_T	4.502875134						
10	λ_x	μ_x	T_x			P_x		$\lambda_x * P_x$	$(x-n) * P(x)$				
11	24	0		1		7.2522E-16		1.74053E-14					
12	22.6667	0.33333333		72		5.22158E-14		1.18356E-12					
13	21.3333	0.66666667		2448		1.77534E-12		3.78739E-11					
14	20	1		52224		3.78739E-11		7.57478E-10					
15	18.6667	1.33333333		783360		5.68108E-10		1.06047E-08					
16	17.3333	1.66666667		8773632		6.36281E-09		1.10289E-07					
17	16	1.66666667		91245772.8		6.61733E-08		1.05877E-06					
18	14.6667	1.66666667		875959418.9		6.35263E-07		9.3172E-06					
19	13.3333	1.66666667		7708442886		5.59032E-06		7.45376E-05					
20	12	1.66666667		61667542000		4.17225E-05		8.00052667					

❖ Hoặc có thể sử dụng OFFSET: vì đã có 1 cột dành cho phép: $\lambda_x * P_x$ nên chỉ cần dùng OFFSET để xác định ô có $x=k$, rồi lấy giá trị đó chia λ_T

I2															
	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1															
2			Output:			$\lambda_x \{x=K\}$	4								
3			$P\{\text{im ser.}\}$	2.53199E-09		$P\{x=K\}$	0.709052117								
4			$P\{\text{del.}\}$	0.370133882		$P\{n-1 < x < K\}$									
5			$P\{\text{loss}\}$	0.629866115		$P\{x < n\}$									
6															
7			$E\{q \text{all state}\}$												
8			$E\{x\}$												
9			sum of Tx	1.37889E+15		λ_T	4.502875134								
10	λ_x	μ_x	T_x			P_x		$\lambda_x * P_x$	$(x-n) * P(x)$						
11	24	0		1		7.2522E-16		1.74053E-14							
12	22.6667	0.33333333		72		5.22158E-14		1.18356E-12							
13	21.3333	0.66666667		2448		1.77534E-12		3.78739E-11							
14	20	1		52224		3.78739E-11		7.57478E-10							
15	18.6667	1.33333333		783360		5.68108E-10		1.06047E-08							
16	17.3333	1.66666667		8773632		6.36281E-09		1.10289E-07							
17	16	1.66666667		91245772.8		6.61733E-08		1.05877E-06							
18	14.6667	1.66666667		875959418.9		6.35263E-07		9.3172E-06							
19	13.3333	1.66666667		7708442886		5.59032E-06		7.45376E-05							
20	12	1.66666667		61667542000		4.17225E-05		8.00052667							

12. Tính $E\{q\}$: độ dài trung bình của hàng đợi

+ Ở tất cả các trạng thái: $E\{q\} = \sum_{x=n+1}^K (x - n) * P(x)$

+ Khi tồn tại hàng đợi ($x > n$): $E\{q|x>n\} = \frac{E\{q\}}{\sum_{x=n+1}^K P(x)}$

- Tính $(x-n)*P(x)$:

Hàng đợi M/M/n/K/N									
Khởi tạo:									
A	4			Output: P{im ser.}	2.53199E-09			$\lambda_x \{x=K\}$	4
n	5			P{del.}	0.370133882			P{x=K}	0.709052117
tm (arrival time)	3			P{loss}	0.629866115			P{n-1 < x < K}	
K	15							P{x < n}	
N (No. of customers)	18			E{q all state}				E{q queue exist}	
λ	1.33			E{x}					
				sum of Tx	1.37889E+15	λ_T	4.502875134		
x	q	λ_x	μ_x	T_x		P_x		$\lambda_x * P_x$	$(x-n) * P(x)$
0	0	24	0	1		7.2522E-16		1.74053E-14	-3.6261E-15
1	0	22.667	0.333333	72		5.22158E-14		1.18356E-12	-2.08863E-13
2	0	21.333	0.666667	2448		1.77534E-12		3.78739E-11	-5.32602E-12
3	0	20	1	52224		3.78739E-11		7.57478E-10	-7.57478E-11
4	0	18.667	1.333333	783360		5.68108E-10		1.06047E-08	-5.68108E-10
5	0	17.333	1.666667	8773632		6.36281E-09		1.10289E-07	0
6	1	16	1.666667	91245772.8		6.61733E-08		1.05877E-06	6.61733E-08
7	2	14.667	1.666667	875959418.9		6.35263E-07		9.3172E-06	1.27053E-06
8	3	13.333	1.666667	7708442886		5.59032E-06		7.45376E-05	1.6771E-05
9	4	12	1.666667	61667543089		4.47225E-05		0.00053667	0.00017889
10	5	10.667	1.666667	4.44006E+11		0.000322002		0.003434691	0.001610011
11	6	9.3333	1.666667	2.84164E+12		0.002060815		0.01923427	0.012364888
12	7	8	1.666667	1.59132E+13		0.011540562		0.092324494	0.080783933
13	8	6.6667	1.666667	7.63833E+13		0.055394697		0.369297978	0.443157573
14	9	5.3333	1.666667	3.05533E+14		0.221578787		1.181753528	1.994209079
15	10	4	1.666667	9.77706E+14		0.709052117		2.836208468	7.090521169
16	11	0	0	0		0		0	0

- Dựa vào công thức để cho:

$$+ Ở tất cả các trạng thái: E{q} = \sum_{x=n+1}^K (x - n) * P(x)$$

SUMIFS(\$H\$11:\$H\$111,\$A\$11:\$A\$111,">="&(B4+1),\$A\$11:\$A\$111, "<="&B6)

với vùng dữ liệu cần tính tổng là từ H11 : H111 kèm 2 điều kiện là chỉ lấy các hàng có $x > n$ và $x \leq K$

Hàng đợi M/M/n/K/N									
Khởi tạo:									
A	4			Output: P{im ser.}	2.53199E-09			$\lambda_x \{x=K\}$	4
n	5			P{del.}	0.370133882			P{x=K}	0.709052117
tm (arrival time)	3			P{loss}	0.629866115			P{n-1 < x < K}	
K	15							P{x < n}	
N (No. of customers)	18			E{q all state}	9.62284365			E{q queue exist}	
λ	1.33			E{x}					
				sum of Tx	1.37889E+15	λ_T	4.502875134		
x	q	λ_x	μ_x	T_x		P_x		$\lambda_x * P_x$	$(x-n) * P(x)$
0	0	24	0	1		7.2522E-16		1.74053E-14	-3.6261E-15
1	0	22.667	0.333333	72		5.22158E-14		1.18356E-12	-2.08863E-13
2	0	21.333	0.666667	2448		1.77534E-12		3.78739E-11	-5.32602E-12
3	0	20	1	52224		3.78739E-11		7.57478E-10	-7.57478E-11
4	0	18.667	1.333333	783360		5.68108E-10		1.06047E-08	-5.68108E-10
5	0	17.333	1.666667	8773632		6.36281E-09		1.10289E-07	0
6	1	16	1.666667	91245772.8		6.61733E-08		1.05877E-06	6.61733E-08
7	2	14.667	1.666667	875959418.9		6.35263E-07		9.3172E-06	1.27053E-06
8	3	13.333	1.666667	7708442886		5.59032E-06		7.45376E-05	1.6771E-05
9	4	12	1.666667	61667543089		4.47225E-05		0.00053667	0.00017889
10	5	10.667	1.666667	4.44006E+11		0.000322002		0.003434691	0.001610011
11	6	9.3333	1.666667	2.84164E+12		0.002060815		0.01923427	0.012364888
12	7	8	1.666667	1.59132E+13		0.011540562		0.092324494	0.080783933
13	8	6.6667	1.666667	7.63833E+13		0.055394697		0.369297978	0.443157573
14	9	5.3333	1.666667	3.05533E+14		0.221578787		1.181753528	1.994209079
15	10	4	1.666667	9.77706E+14		0.709052117		2.836208468	7.090521169
16	11	0	0	0		0		0	0

$$+ Khi tồn tại hàng đợi (x>n): E{q|x>n} = \frac{E{q}}{\sum_{x=n+1}^K P(x)}$$

F7/SUMIFS(\$F\$11:\$F\$111,\$A\$11:\$A\$111,">="&(B4+1),\$A\$11:\$A\$111, "<="&B6)

với F7 là E{q} đã tính chia cho SUMIFS(...)

	B	C	D	E	F	G	H	I	J	K
1										
2				Output:						
3	4			P{im ser.}	2.53199E-09		$\lambda_x \{x=K\}$		4	
4	5			P{del.}	0.370133882		$P\{x=K\}$	0.709052117		
5	3			P{loss}	0.629866115		$P\{n-1 < x < K\}$			
6	15						$P\{x < n\}$			
7	18			E{q all state}	9.62284365		E{q queue exist}	9.622843717		
8	1.33			E{x}						
9			sum of Tx	1.37889E+15	λ_T	4.502875134				
10	q	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	$(x-n) * P(x)$			
11	0	24	0	1	7.2522E-16	1.74053E-14	-3.6261E-15			
12	0	22.667	0.333333	72	5.22158E-14	1.18356E-12	-2.08863E-13			
13	0	21.333	0.666667	2448	1.77534E-12	3.78739E-11	-5.32602E-12			
14	0	20	1	52224	3.78739E-11	7.57478E-10	-7.57478E-11			
15	0	18.667	1.333333	783360	5.68108E-10	1.06047E-08	-5.68108E-10			
16	0	17.333	1.666667	8773632	6.36281E-09	1.10289E-07	0			
17	1	16	1.666667	91245772.8	6.61733E-08	1.05877E-06	6.61733E-08			
18	2	14.667	1.666667	875959418.9	6.35263E-07	9.3172E-06	1.27053E-06			
19	3	13.333	1.666667	7708442886	5.59032E-06	7.45376E-05	1.6771E-05			
20	4	12	1.666667	61667543089	4.47225E-05	0.00053667	0.00017889			
21	5	10.667	1.666667	4.44006E+11	0.000322002	0.003434691	0.001610011			
22	6	9.3333	1.666667	2.84164E+12	0.002060815	0.01923427	0.012364888			
23	7	8	1.666667	1.59132E+13	0.011540562	0.092324494	0.080783933			
24	8	6.6667	1.666667	7.63833E+13	0.055394697	0.369297978	0.443157573			
25	9	5.3333	1.666667	3.05533E+14	0.221578787	1.181753528	1.994209079			
26	10	4	1.666667	9.77706E+14	0.709052117	2.836208468	7.090521169			

13. Tính luôn các ô chưa có kết quả

- $$- P\{n-1 < x < K\}:$$

SUMIFS(\$F\$11:\$F\$111, \$A\$11:\$A\$111, ">=" & \$B\$4, \$A\$11:\$A\$111, "<" & \$B\$6)

	A	B	C	D	E	F	G	H	I	J
1	Hàng đợi M/M/n/K/N									
2	Khởi tạo:				Output:			$\lambda_x \{x=K\}$		4
3	A	4			P{im ser.}	2.53199E-09		P{x=K}		0.709052117
4	n	5			P{del.}	0.370133882		P{n-1 < x < K}		0.290947883
5	tm (arrival time)	3			P{loss}	0.629866115		P{x < n}		6.07811E-10
6	K	15								
7	N (No. of customers)	18			E{q all state}	9.62284365		E{q queue exist}		9.622843717
8	λ	1.33			E{x}					
9				sum of Tx	1.37889E+15		λT	4.502875134		
10	x	q	λ_x	μ_x	T.x	P.x	$\lambda_x * P.x$	$(x-n) * P(x)$		
11	0	0	24	0	1	7.2522E-16	1.74053E-14	-3.6261E-15		
12	1	0	22.667	0.333333	72	5.22158E-14	1.18356E-12	-2.08863E-13		
13	2	0	21.333	0.666667	2448	1.77534E-12	3.78739E-11	-5.32602E-12		
14	3	0	20	1	52224	3.78739E-11	7.57478E-10	-7.57478E-11		
15	4	0	18.667	1.333333	783360	5.68108E-10	1.06047E-08	-5.68108E-10		

- P{x<n}: =SUMIFS(\$F\$11:\$F\$111, \$A\$11:\$A\$111, "<"&\$B\$4)

	Clipboard	Font	Alignment	Number	Styles	Cells
I5	<input type="button" value="X"/> <input type="button" value="✓"/> <input type="button" value="fx"/>	=SUMIFS(\$F\$11:\$F\$111, \$A\$11:\$A\$111, "<"&\$B\$4)				
A	B	C	D	E	F	G
1 Hàng đợi M/M/n/K/N						H
2 Khởi tạo:				Output:		
3 A	4			P{im ser.}	2.53199E-09	P{x=K}
4 n	5			P{del.}	0.370133882	P{n-1<x<K}
5 tm (arrival time)	3			P{loss}	0.629866115	P{x<n}
6 K	15					0.709052117
7 N (No. of customers)	18			E{q all state}	9.62284365	0.290947883
8 λ	1.33			E{x}		6.07811E-10
9			sum of Tx	1.37889E+15	λ_T	4.502875134
10 x	q	λ_x	μ_x	T_x	P_x	$(x-n)*P(x)$
11 0	0	24	0	1	7.2522E-16	1.74053E-14
12 1	0	22.667	0.3333333	72	5.22158E-14	1.18356E-12
13 2	0	21.333	0.666667	2448	1.77534E-12	3.78739E-11
14 3	0	20	1	52224	3.78739E-11	5.32602E-12
					7.57478E-10	-7.57478E-11

14. Thay đổi các thông số bao gồm A, K, N để ra các trường hợp sau:

- Hệ thống hàng đợi rảnh và cường độ thấp: tìm A và N với điều kiện $A < 1$ và $n < N \leq K$ sao cho xác xuất $P\{\text{immediate service}\}$ tiến về ~1 và $P\{\text{loss}\} = 0$
- Hệ thống hàng đợi chưa bị quá tải và cường độ cao: tìm A và N với điều kiện $A > 1$ và $n < N \leq K$ sao cho xác xuất $P\{\text{immediate service}\}$ tiến về ~1 và $P\{\text{loss}\} = 0$

Hãy thay đổi và giải thích từng trường hợp trên.

Nhắc lại công thức:

$$P\{\text{immediate service}\} = \sum_{x=0}^{n-1} \frac{\lambda_x * P_x}{\lambda_T}$$

a) Hệ rảnh và cường độ thấp:

- Điều kiện:** $A < 1$, $n < N \leq K$ sao cho $P\{\text{immediate service}\} \rightarrow \sim 1$ và $P\{\text{loss}\} = 0$.
- Ta có: $A = \text{offered traffic} = \lambda * t_m$
 - Đặt $A = 0,5$ (thỏa $A < 1$) \rightarrow chọn $\lambda = 0,5 \rightarrow t_m = 1$
 - Giữ $n = 5$, đặt $N = 10$, giữ $K = 15$ (thỏa $n < N \leq K$)

A	B	C	D	E	F	G	H	I
1 Hàng đợi M/M/n/K/N								
2 Khởi tạo:				Output:				
3 A	0.5			P\{\text{im ser.}\}	0.843728626	$\lambda_x \{x=K\}$	-2.5	
4 n	5			P\{\text{del.}\}	0.156271374	P\{x=K\}	0	
5 t_m (arrival time)	1			P\{\text{loss}\}	0	P\{n-1 < x < K\}	0.239358953	
6 K	15					P\{x < n\}	0.777782254	
7 N (No. of custumers)	10			E\{q all state\}	0.153075261			
8 λ	0.50			E\{x\}		E\{q queue exist\}	1.466632178	
9				sum of Tx	58.33895	$\lambda * T$	3.339445602	
10	x	q	λ_x	μ_x	T_x	P_x	$\lambda_x * P_x$	$(x * n) * P(x)$
11	0	0	5	0	1	0.017141207	0.085706033	-0.085706033
12	1	0	4.5	1	5	0.085706033	0.385677151	-0.342824134
13	2	0	4	2	11.25	0.192838575	0.771354301	-0.578515726
14	3	0	3.5	3	15	0.2571181	0.899913351	-0.514236201
15	4	0	3	4	13.125	0.224979338	0.674935013	-0.224979338
16	5	0	2.5	5	7.875	0.134987003	0.337467507	0
17	6	1	2	5	3.9375	0.067493501	0.134987003	0.067493501
18	7	2	1.5	5	1.575	0.026997401	0.040496101	0.053994801
19	8	3	1	5	0.4725	0.00809922	0.00809922	0.02429766
20	9	4	0.5	5	0.0945	0.001619844	0.000809922	0.006479376
21	10	5	0	5	0.00945	0.000161984	0	0.000809922
22	11	6	-0.5	5	0	0	0	0

Kết quả thu được: $P\{\text{immediate service}\} \rightarrow \sim 1$ và $P\{\text{loss}\} = 0$

Giải thích:

- $A < 1$ nghĩa là tổng công việc **nhỏ hơn** khả năng xử lý một server trong đơn vị thời gian \rightarrow Nếu có nhiều server thì hệ sẽ rảnh
- Nếu n không quá nhỏ so với A, hầu hết thời gian sẽ có ít khách hơn n \rightarrow ngay khi arrival đến sẽ có server rảnh \rightarrow immediate service ≈ 1
- K lớn làm giảm xác suất rơi vào trạng thái đầy:** Nếu K đủ lớn (hoặc vô hạn), thì khả năng đạt trạng thái đầy $x = K$ rất nhỏ $\rightarrow P\{\text{loss}\} \approx 0$
- Nếu $n < N \leq K \rightarrow$** số khách tiềm năng lớn hơn số server, nhưng do A nhỏ, vẫn không xảy ra queue đầy
- Với $A = \lambda * t_m = 0.5$ và $n = 5$,** tỷ lệ tải trên mỗi server $\rho = A/n = 0.1 \Leftrightarrow$ rất nhỏ. Do đó hầu hết thời gian số khách trong hệ nhỏ hơn n, vì vậy $P(X < n)$ tiến gần 1.
- Vì K được chọn lớn ($K = 15$) so với dao động, xác suất đạt trạng thái đầy $P(X = K)$ xấp xỉ 0,** tức không có mất mát.

b) Hệ chưa bị quá tải nhưng cường độ cao:

❖ **Điều kiện:** $A > 1$, $n < N \leq K$ sao cho $P\{\text{immediate service}\} \rightarrow \sim 1$ và $P\{\text{loss}\} = 0$.

- Ta có: $A = \text{offered traffic} = \lambda * t_m$
 - Đặt $A = 10$ (thỏa $A > 1$) \rightarrow chọn $\lambda = 10 \rightarrow t_m = 1$
 - Chọn $n = 30$, $N = 40$, $K = 50$ (thỏa $n < N \leq K$)

	A	B	C	D	E	F	G	H	I	J
1	Hàng đợi M/M/n/K/N									
2	Khởi tạo:				Output:					
3	A	10			$P\{\text{im ser.}\}$	0.00108624	$\lambda_x\{x=K\}$		-100	
4	n	30			$P\{\text{del.}\}$	0.99891376	$P\{x=K\}$	0		
5	t_m (arrival time)	1			$P\{\text{loss}\}$	0	$P\{n-1 < x < K\}$	0.999711479		
6	K	50					$P\{x < n\}$	0.000288521		
7	N (No. of customers)	40			$E\{q \text{all state}\}$	7.000410818	$E\{q \text{queue exist}\}$	7.008110453		
8	λ	10.00			$E\{x\}$					
9					$\sum \text{of Tx}$	1.0463E+42	λ_T	29.99962653		
10	x	q	λ_x	μ_x	T_x		P_x	$\lambda_x * P_x$	$(x-n) * P(x)$	
11	0	0	400	0	1	9.55753E-43	3.82301E-40	-2.86726E-41		
12	1	0	390	1	400	3.82301E-40	1.49098E-37	-1.10867E-38		
13	2	0	380	2	78000	7.45488E-38	2.83285E-35	-2.08737E-36		
14	3	0	370	3	9880000	9.44284E-36	3.49385E-33	-2.54957E-34		
15	4	0	360	4	913900000	8.73463E-34	3.14447E-31	-2.271E-32		
16	5	0	350	5	65800800000	6.28893E-32	2.20113E-29	-1.57223E-30		
17	6	0	340	6	3.83838E+12	3.66854E-30	1.2473E-27	-8.80451E-29		
18	7	0	330	7	1.86436E+14	1.78186E-28	5.88015E-26	-4.09829E-27		
19	8	0	320	8	7.69047E+15	7.35019E-27	2.35206E-24	-1.61704E-25		
20	9	0	310	9	2.73439E+17	2.6134E-25	8.10154E-23	-5.48814E-24		
21	10	0	300	10	8.47661E+18	8.10154E-24	2.43046E-21	-1.62031E-22		
22	11	0	290	11	2.3118E+20	2.20951E-22	6.40758E-20	-4.19807E-21		
23	12	0	280	12	5.58685E+21	5.33965E-21	1.4951E-18	-9.61138E-20		

❖ **Kết quả thu được:** $P\{\text{immediate service}\} \rightarrow \sim 1$ và $P\{\text{loss}\} = 0$

❖ **Giải thích:**

- $A > 1$ nghĩa là tổng lượng công việc lớn hơn khả năng 1 server, nhưng không qui định về số server n \rightarrow Nếu n lớn so với A, tức $\rho = A/n$ vẫn nhỏ, hệ vẫn có nhiều server rảnh mặc dù tổng A > 1.

\Leftrightarrow Do đó thấy được yếu tố quyết định immediate service là mối quan hệ giữa A và n, chứ không chỉ A đơn độc.

- Tăng n: thêm server để chia tải từ đó làm cho $\rho = A/n$ nhỏ
- **K lớn** đảm bảo không bị **overflow** \rightarrow tránh $P\{\text{loss}\} > 0$
- **N đủ lớn** để mô phỏng nguồn khách “gần vô hạn”, đảm bảo **arrival đều đặn** \rightarrow hệ vận hành ở công suất cao nhưng vẫn ổn định.

\Rightarrow Mặc dù tổng offered traffic A=10 là lớn, nhưng khi tăng số server lên n=30 thì tải trung bình mỗi server là $\rho = A/n \approx 0.333$.

\Rightarrow Vì ρ vẫn ở mức thấp đến trung bình, hầu hết thời gian hệ có ít hơn n khách và arrival hầu như luôn gấp server rảnh. Nếu K đủ lớn thì sẽ đạt được kết quả mong muốn

⊕ **Tùy 2 ví dụ trên suy ra được vai trò của K và N**

- ✓ **K:** giới hạn tối đa số khách trong hệ.
 - Khi K nhỏ, nguy cơ loss tăng vì sẽ dễ bị trạng thái đầy
 - Khi K lớn (hoặc vô hạn) thì $P(X=K) \rightarrow 0$.

$\Rightarrow K$ kiểm soát giới hạn vật lý (bộ nhớ...), tăng K làm giảm xác suất loss nhưng có chi phí hạ tầng.
- ✓ **N:** nếu nguồn khách hữu hạn (N nhỏ), khi nhiều khách đã trong hệ, arrival effective giảm (vì số khách ngoài hệ còn lại ít) \rightarrow giảm khả năng quá tải so với nguồn vô hạn. Vì vậy nếu N nhỏ (gần n) thì hệ ít có khả năng tràn \rightarrow giảm loss.

$\Rightarrow N$ biểu diễn kích thước nguồn; khi N nhỏ, arrival giảm khi hệ đầy \rightarrow làm giảm tắc.

✚ Đánh giá khả năng mở rộng và giải pháp tối ưu hệ thống hàng đợi trong thực tế

Giải pháp	Mức độ	Ứng dụng thực tế
Tăng n	Trung bình - khó (tốn tài nguyên)	Dễ mở rộng trong hệ thống máy chủ, khó trong hệ thống vật lý
Giữ K lớn	Dễ	Dễ thực hiện nhưng chỉ cải thiện tạm thời
Giảm N	Khó	Hữu ích trong hệ thống kỹ thuật số có thể kiểm soát luồng truy cập

15. Với thông số mặc định trong file excel thay $A = 3$, hãy giải thích vì sao khi $N = n = 5$ thì hệ thống đợi có $P\{\text{immediate service}\} = 1$ và khi bằng đầu tăng $N = 6,7,8,9,\dots, K$ thì $P\{\text{immediate service}\}$ lại tăng nhanh

✚ Đổi $A = 3, N = n = 5$:

A	B	C	D	E	F	G	H	I	J
1 Hàng đợi M/M/n/K/N									
2 Khởi tạo:				Output:					
3 A	3			$P\{\text{im ser.}\}$	1		$\lambda_x \{x=K\}$		-10
4 n	5			$P\{\text{del.}\}$	0		$P\{x=K\}$	0	
5 tm (arrival time)	3			$P\{\text{loss}\}$	0		$P\{n-1 < x < K\}$	0.237536657	
6 K	15						$P\{x < n\}$	0.76344086	
7 N (No. of custumers)	5			$E\{q \text{all state}\}$	0				
8 λ	1.00			$E\{x\}$			$E\{q \text{queue exist}\}$	#DIV/0!	
				sum of Tx	1023	λ_T	1.251221896		
10 x	q	λ_x	μ_x	T_x		P_x	$\lambda_x * P_x$	$(x-n) * P(x)$	
11 0	0	5	0	1	0.000977517	0.004887586	-0.004887586		
12 1	0	4	0.3333333	15	0.014662757	0.058651026	-0.058651026		
13 2	0	3	0.6666667	90	0.08797654	0.263929619	-0.263929619		
14 3	0	2	1	270	0.263929619	0.527859238	-0.527859238		
15 4	0	1	1.3333333	405	0.395894428	0.395894428	-0.395894428		
16 5	0	0	1.6666667	243	0.237536657	0	0		
17 6	1	-1	1.6666667	0	0	0	0	0	

→ $P\{\text{immediate service}\} = 1$

Giải thích:

Nếu tổng số nguồn khách ngoài hệ bằng đúng số server ($N = n = 5$), thì **tại thời điểm tất cả server đang bận (x = n)**, **không còn khách nào ở ngoài** để phát sinh arrival nữa → arrival rate tại trạng thái đầy $\lambda_n = 0$.

$$\lambda_n = \lambda_{\text{ind}} * (N - n) = \lambda_{\text{ind}} * 0 = 0$$

Do đó, **arrival chỉ có thể xảy ra khi hệ đang ở trạng thái có ít hơn n khách (x < n)** — tức là **luôn có ít nhất một server rảnh** tại thời điểm khách mới đến.

Khi dùng định nghĩa **arrival-weighted cho $P\{\text{immediate}\}$** , vì $\lambda_n=0$ nên tỉ số giữa “arrival diễn ra khi có server rảnh” và “tất cả arrival” = 1

→ **SUY RA:** khi $N = n$ thì mọi arrival đều luôn gặp server rảnh $\Leftrightarrow P\{\text{immediate}\} = 1$

✚ Đổi $N = 6,7,8,\dots,K$

N	6	7	8	9	10	11	12	13	14	15
$P\{\text{i.s}\}$	0.763	0.45	0.211	0.081	0.026	0.007	0.002	0.0003	5.9E-05	9.7E-06

→ Càng tiến về 0

Giải thích:

N tăng, có nhiều người “bên ngoài” hơn, với $n=5, N=6 \rightarrow 15$, điều này có nghĩa là vẫn **có khách đến ngay cả khi tất cả server bận** - những arrivals này sẽ không được phục vụ ngay, do đó kéo $P\{\text{immediate}\}$ xuống

Khi N lớn hơn nhiều so với n , các giá trị ($N-x$) cho những trạng thái gần $x=n, n+1, \dots$ trở nên lớn, làm cho tốc độ hiệu dụng lamda_x tăng mạnh cho nên **mẫu số càng ngày càng lớn so với tử số** dẫn đến $P\{i\}$ càng ngày càng giảm

Phân phối trạng thái Px dịch hướng sang các trạng thái lớn ($x \geq n$): khi arrival rate hiệu dụng tăng, hệ có xu hướng chứa nhiều khách hơn. Khi xác suất tồn tại ở trạng thái $x \geq n$ (đầy hoặc gần đầy) tăng, thì phần tử tử số chiếm tỉ lệ nhỏ hơn trong tổng λT

Hiệu ứng phi tuyến: Đồng thời, P_x cho $x \geq n$ có thể tăng nhanh (hiệu ứng tích lũy), tạo vòng lặp dương: càng nhiều arrival \rightarrow hệ càng thường xuyên ở trạng thái đầy \rightarrow lại càng có nhiều arrival tại các trạng thái đầy \rightarrow mẫu số càng áp đảo so với tử số \rightarrow giảm rất nhanh
(Đây chính là **hiệu ứng phi tuyến của hệ birth-death: một thay đổi nhỏ ở N cũng có thể khuếch đại mạnh trong phân phối trạng thái, làm biến thiên kết quả đầu ra theo hướng phi tuyến**)

A		B		C		D		E		F		A		B		C		D		E		F		
Hàng đợi M/M/n/K/N		Output:		Hàng đợi M/M/n/K/N		Output:		Hàng đợi M/M/n/K/N		Output:		Hàng đợi M/M/n/K/N		Output:		Hàng đợi M/M/n/K/N		Output:		Hàng đợi M/M/n/K/N				
Khởi tạo:		Output:		Khởi tạo:		Output:		Khởi tạo:		Output:		Khởi tạo:		Output:		Khởi tạo:		Output:		Khởi tạo:		Output:		
A	3	P{im ser.}	0.762695313	A	3	P{im ser.}	0.211256124	n	5	P{del.}	0.237304688	n	5	P{del.}	0.788743876	tm (arr)	3	P{loss}	0	tm (arr)	3	P{loss}	0	
n	5			n	5			tm (arr)	3			tm (arr)	3			K	15			K	15			
tm (arr)	3			tm (arr)	3			K	15			K	15											
K	15																							
N (No. o	6			E{q all state}	0.206281834			N (No. o	8			E{q all state}	1.470886508			E(x)	1.00			E(x)	1.00			
λ	1.00							λ	1.00															
sum of Tx		4240.8		$\lambda \cdot T$		sum of Tx		92589.648		$\lambda \cdot T$		sum of Tx		92589.648		$\lambda \cdot T$		sum of Tx		$\lambda \cdot T$				
x	q	$\lambda \cdot x$	$\mu \cdot x$	T_x	P_x	x	q	$\lambda \cdot x$	$\mu \cdot x$	T_x	P_x	x	q	$\lambda \cdot x$	$\mu \cdot x$	T_x	P_x	x	q	$\lambda \cdot x$	$\mu \cdot x$	T_x	P_x	
0	0	6	0	1	0.000235805	0	0	8	0	0	1	0	0	7	0	0.33333333	24	0.000259208	0	0	0	0	1	1.08003E-05
1	0	5	0.33333333	18	0.004244482	1	0	6	0.66666667	252	0.002721687	2	0	5	1	1512	0.016330119	0	0	0	0	0	0	
2	0	4	0.66666667	135	0.031833616	3	0	4	0.13333333	5670	0.061237947	4	0	3	1.66666667	13608	0.146971074	0	0	0	0	0	0	
3	0	3	1	540	0.127334465	5	0	3	1.66666667	244944	0.264547933	6	1	2	1.66666667	29393.28	0.31745752	0	0	0	0	0	0	
4	0	2	1.33333333	1215	0.286502547	7	2	1	1.66666667	17635.968	0.190474512	8	0	0	0	0	0	9	4	-1	1.66666667	0	0	0
5	0	1	1.66666667	1458	0.343803056	10	5	-4	1.66666667	0	0	11	6	-3	1.66666667	0	0	12	7	-4	1.66666667	0	0	0
6	1	0	1.66666667	874.8	0.206281834	13	8	-5	1.66666667	0	0	14	9	-6	1.66666667	0	0	15	10	-7	1.66666667	0	0	0
7	2	-1	1.66666667	0	0	16	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	3	-2	1.66666667	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
9	4	-3	1.66666667	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10	5	-4	1.66666667	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
11	6	-5	1.66666667	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
12	7	-6	1.66666667	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
13	8	-7	1.66666667	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
14	9	-8	1.66666667	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
15	10	-9	1.66666667	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
16	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

A		B		C		D		E		F		A		B		C		D		E		F		
Hàng đợi M/M/n/K/N		Output:		Hàng đợi M/M/n/K/N		Output:		Hàng đợi M/M/n/K/N		Output:		Hàng đợi M/M/n/K/N		Output:		Hàng đợi M/M/n/K/N		Output:		Hàng đợi M/M/n/K/N				
Khởi tạo:		Output:		Khởi tạo:		Output:		Khởi tạo:		Output:		Khởi tạo:		Output:		Khởi tạo:		Output:		Khởi tạo:		Output:		
A	3	P{im ser.}	0.001588467	A	3	P{im ser.}	9.74287E-06	n	5	P{del.}	0.99990257	n	5	P{del.}	5.30825E-10	tm (arr)	3	P{loss}	0	tm (arr)	3	P{loss}	0	
n	5			n	5			tm (arr)	3			tm (arr)	3			K	15			K	15			
tm (arr)	3			tm (arr)	3			K	15			K	15			E(q all state)	8.333335506			E(q all state)	8.333335506			
K	15							λ	1.00			λ	1.00											
N (No. o	12			E(q all state)	5.333333334																			
λ	1.00			E(x)	0																			
sum of Tx		143759946.7		$\lambda \cdot T$		sum of Tx		84773723092		$\lambda \cdot T$		sum of Tx		84773723092		$\lambda \cdot T$		sum of Tx		$\lambda \cdot T$				
x	q	$\lambda \cdot x$	$\mu \cdot x$	T_x	P_x	x	q	$\lambda \cdot x$	$\mu \cdot x$	T_x	P_x	x	q	$\lambda \cdot x$	$\mu \cdot x$	T_x	P_x	x	q	$\lambda \cdot x$	$\mu \cdot x$	T_x	P_x	
0	0	12	0	1	6.95604E-09	0	0	15	0	0	1	0	0	14	0	0.33333333	45	5.11473E-08	0	0	12	1	1.17961E-11	
1	0	11	0.33333333	36	2.50417E-07	1	0	13	0.66666667	945	1.144915E-07	2	0	12	1	12285	110565	1.30424E-06	0	0	11	4	6.07096E-06	
2	0	10	0.66666667	594	4.13189E-06	3	0	12	1	1.33333333	110565	4	0	11	1	1.33333333	110565	1.30424E-06	0	0	10	5	4.378374	
3	0	9	1	5940	4.13189E-05	5	0	10	1.66666667	729729	8.60796E-05	6	1	9	1.66666667	4378374	5.16478E-05	0	0	9	6	23643219.6		
4	0	8	1.33333333	40095	0.000278902	7	0	8	1.66666667	113487454.1	0.000278898	8	3	7	1.66666667	113487454.1	0.000133871	0	0	8	7	0		
5	0	7	1.66666667	192456	0.001338722	9	4	6	1.66666667	476647307.1	0.005622583	10	5	5	1.66666667	171593036	0.020241299	0	0	7	4	945		
6	1	6	1.66666667	808315.2	0.005622673	11	6	4	1.66666667	5147790917	0.060723898	12	7	3	1.66666667	12354698201	0.145737355	0	0	6	1	4378374		
7	2	5	1.66666667	2909934.72	0.020241624	13	8	2	1.66666667	22238456762	0.26232724	14	9	1	1.66666667	26686148114	0.314792687	0	0	5	7	0		
8	3	4	1.66666667	8729804.16	0.060724871	15	10	0	1.66666667	16011688868	0.188875612	16	11	0	0	0	0	0	0	0	0	0	0	
9	4	3	1.66666667	20951529.98	0.145739689	17	12	7	1.66666667	12354698201	0.145737355	18	13	6	1.66666667	22238456762	0.26232724	19	14	5	1.66666667	0		
10	5	2	1.66666667	37712753.97	0.262331441	20	15	4	1.66666667	16011688868	0.188875612	21	16	3	1.66666667	16011688868	0.188875612	22	17	2	1			

YÊU CẦU CHUNG

1) Đánh giá

- Chuẩn bị tốt các yêu cầu đặt ra trong bài thực hành.
- Sinh viên hiểu và tự thực hiện được bài thực hành, trả lời đầy đủ các yêu cầu đặt ra.
- Nộp báo cáo kết quả chi tiết những đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (*nếu có*); giải thích cho quan sát (*nếu có*).
- Sinh viên báo cáo kết quả thực hiện và nộp bài.

2) Báo cáo

- File .PDF hoặc .docx. Tập trung vào nội dung, giải thích.
- Nội dung trình bày bằng Font chữ Times New Romans/ hoặc font chữ của mẫu báo cáo này (UTM Avo)– cỡ chữ 13. Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.
- Đặt tên theo định dạng: LabX_MSSV1_MSSV2. (trong đó X là Thứ tự buổi Thực hành).
Ví dụ: Lab01_21520001_21520002
- Nộp file báo cáo trên thời gian đã thống nhất tại courses.uit.edu.vn.
Bài sao chép, trẽ, ... sẽ được xử lý tùy mức độ vi phạm.

HẾT