Operational Research / Queuing Systems
Practice Set 1

Problem 1

Plot the queuing processes $N(t)$ = *queue length* and $U(t)$ = *unfinished work* for an arrival stream specified by (t_n^+) = (0.5 s, 2 s, 5 s, 6.5 s) and (b_n) =(3 s.u., 2 s.u., 2 s.u., 3 s.u.), and for two cases:

(a) there are two processors each serving requests at the speed of v = 1 s.u./s (no grading, processor-bound and time-sharing service mode are assumed),
(b) there is one processor serving requests at the speed of v = 2 s.u./s (time-sharing service mode is assumed).

Compare the two cases from various viewpoints.

Be careful with the slopes of $U(t)$! What are the merits of case (b)? Is case (a) superior from some point of view?

Problem 2

Compare mean system delays in a single-processor queuing system under FIFO and RR with service quantum 2 s (partial use of assigned quantum causes earlier commencement of the next quantum). Processor speed is 1 s.u./s. Three requests, X, Y and Z, of sizes 7 s.u., 1 s.u. and 3 s.u., respectively, arrive simultaneously and queue up in the order (a) XYZ, (b) YZX, (c) XZY.

Can any general properties of FIFO and RR be inferred from such a limited set of scenarios?

Problem 3

A queuing system serves on average 800 transactions per second, each transaction on average requiring 5000 elementary operations to complete. An arriving transaction is immediately assigned a processor whose speed is 4,000,000 elementary operations/s. Find the mean number of transactions in system.

Use Little's theorem.

Operational Research / Queuing Systems
Practice Set 2

w każdej 1s wpływa popyt na 750 ms

Problem 1

Gęsty strumień

A single-processor infinite-buffer queuing system with processor speed $v = 1$ s.u./s serves a "dense" arrival stream of requests creating a 75% offered load. The total service demand in a one-second observation period is a random variable with standard deviation $\sigma = 0.1$ s. What are the chances that the processor can spare half of the second to deal with other (e.g., system) tasks without a backlog of requests forming at the end of the observation period?

Use the **Laplace function**: $\Phi(x) = \frac{1}{\sqrt{2\pi}}\int_0^x e^{-y^2/2}dy$. Why Laplace function?

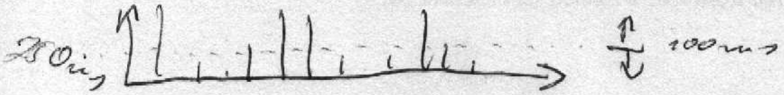
Tables of the Laplace function are widely available. Some characteristic values: $\Phi(0.5) = 0.191$, $\Phi(1) = 0.341$, $\Phi(1.5) = 0.433$, $\Phi(2) = 0.477$, $\Phi(2.5) = 0.4938$, $\Phi(3) = 0.4987$, $\Phi(\geq 4) \approx 0.5$. What to do if $x < 0$?

Problem 2

A single-processor queuing system with a finite buffer of capacity Q works under offered load $r > 1$. In such a system, the loss fraction L never drops below a certain level. What level is this?

Recall the flow conservation equation. How often is the processor idle if $Q \to \infty$?

ciągłe przepływy



Musimy mieć 0.5 s wolnego

czy to się uda, jeśli jest tego co ile sekund?

czy i jak często praca będzie w

rozkład normalny

mamy średnią

i odchylenie standardowe

Problem 3

A processor handles telemetric reports generated by a number of identical sources. To enable multiple access, the processor is equipped with a finite buffer that can accommodate up to Q reports. The table below contains the mean system delay of a report, normalized to the mean report processing time (in boldface), and report loss fraction due to buffer overflow, as dependent on Q and offered load r. Find the maximum number of sources that can be connected to the processor and necessary buffer capacity under the following assumptions:

- each source generates on average 20 reports per minute,
- a report contains on average 1800 records of data,
- the processor is capable of handling 12000 records per second,
- tolerable mean system delay of a report is 1.8 s,
- tolerable loss fraction due to buffer overflow is 4%.

znormalizowane!

normalizowany średni dosyły systemu

ułamek utraconych zgłoszeń.

Q= *r* =	20		21		22		23		24		25	
0.1	**1.11**	0	**1.11**	0	**1.11**	0	**1.11**	0	**1.11**	0	**1.11**	0
0.2	**1.25**	0	**1.25**	0	**1.25**	0	**1.25**	0	**1.25**	0	**1.25**	0
0.3	**1.43**	0	**1.43**	0	**1.43**	0	**1.43**	0	**1.43**	0	**1.43**	0
0.4	**1.67**	0	**1.67**	0	**1.67**	0	**1.67**	0	**1.67**	0	**1.67**	0
0.5	**2**	0	**2**	0	**2**	0	**2**	0	**2**	0	**2**	0
0.6	**2.5**	0	**2.5**	0	**2.5**	0	**2.5**	0	**2.5**	0	**2.5**	0
0.7	**3.32**	0	**3.32**	0	**3.32**	0	**3.33**	0	**3.33**	0	**3.33**	0
0.8	**4.77**	0	**4.8**	0	**4.84**	0	**4.86**	0	**4.89**	0	**4.91**	0
0.9	**7.23**	0.01	**7.42**	0.01	**7.6**	0.01	**7.76**	0.01	**7.92**	0.01	**8.07**	0.01
1	**10.5**	0.05	**11**	0.05	**11.5**	0.04	**12**	0.04	**12.5**	0.04	**13**	0.04
1.1	**13.5**	0.11	**14.3**	0.1	**15.1**	0.1	**15.9**	0.1	**16.7**	0.1	**17.5**	0.1
1.2	**15.5**	0.17	**16.5**	0.17	**17.4**	0.17	**18.4**	0.17	**19.3**	0.17	**20.3**	0.17
1.3	**16.8**	0.23	**17.8**	0.23	**18.7**	0.23	**19.7**	0.23	**20.7**	0.23	**21.7**	0.23
1.4	**17.5**	0.29	**18.5**	0.29	**19.5**	0.29	**20.5**	0.29	**21.5**	0.29	**22.5**	0.29
1.5	**18**	0.33	**19**	0.33	**20**	0.33	**21**	0.33	**22**	0.33	**23**	0.33
1.6	**18.3**	0.38	**19.3**	0.38	**20.3**	0.38	**21.3**	0.38	**22.3**	0.38	**23.3**	0.38
1.7	**18.6**	0.41	**19.6**	0.41	**20.6**	0.41	**21.6**	0.41	**22.6**	0.41	**23.6**	0.41
1.8	**18.8**	0.44	**19.8**	0.44	**20.8**	0.44	**21.8**	0.44	**22.8**	0.44	**23.8**	0.44
1.9	**18.9**	0.47	**19.9**	0.47	**20.9**	0.47	**21.9**	0.47	**22.9**	0.47	**23.9**	0.47
2	**19**	0.5	**20**	0.5	**21**	0.5	**22**	0.5	**23**	0.5	**24**	0.5

$d \leq 12$

$L \leq 0{,}04$

więcej końcowe

Operational Research / Queuing Systems
Practice Set 3

nadajnik to procesor

Problem 1

Each of 50 terminals connected to a common transceiver generates a request after a think time of average duration $^2/_3$ s. In 80% cases it is a message of average length 1000 bytes, and in 20% cases a control data report of average length 160 bytes. The transceiver works at 1 Mb/s in half-duplex; the average proportion of time it is switched to receive mode is 75% (during that time it is unavailable to the terminals). What is the resulting loss fraction?

Data in the problem permit to identify a_m, b_m and proportion of processor idle time. Now go back to the flow conservation equation...

Problem 2

In a single-processor queuing system with buffer capacity $Q = 2$ in statistical equilibrium and under offered load $r = 0.75$, we have $p_0 \geq p_1 \geq p_2$. Find the range of possible p_1 values.

Proceed as in the previous problem.

Problem 3 (simulation experiment)

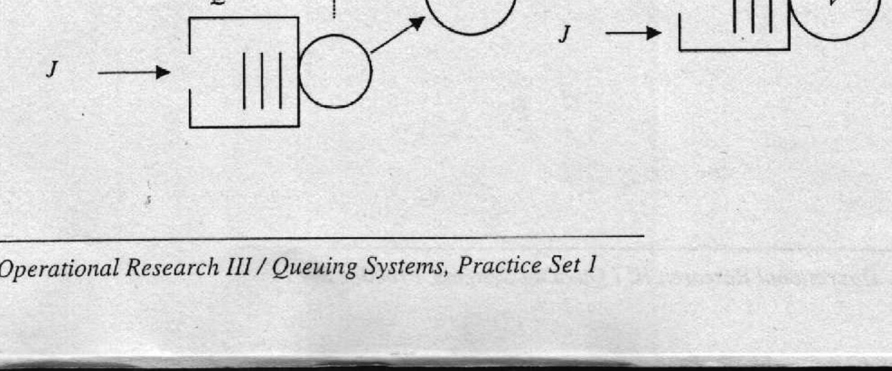
Each user of a single-processor queuing system generates an arrival stream of documents with mean interval a_m s. Documents have a mean length of b_m bytes. The processor handles documents at speed v bytes/s. Tolerable are:

- document loss fraction due to buffer overflow not exceeding L_{max}
- mean system delay of a document not exceeding a given multiple c of b_m/v.

Subject to the above, compare the maximum number J_{max} of users and required buffer capacity in two configurations:

(a) *dedicated access* with a virtual processor and a separate buffer space assigned for each user, and
(b) *common-buffer access* to the processor.

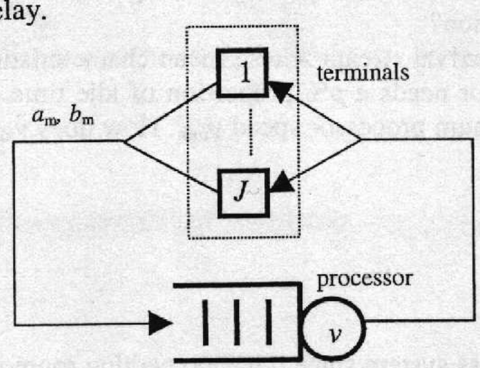
Perform simulations for $a_m = 6$ s, $b_m = 600$ bytes, $v = 24000$ bytes/s, $L_{max} = 0.1\%$, $c = 5$.



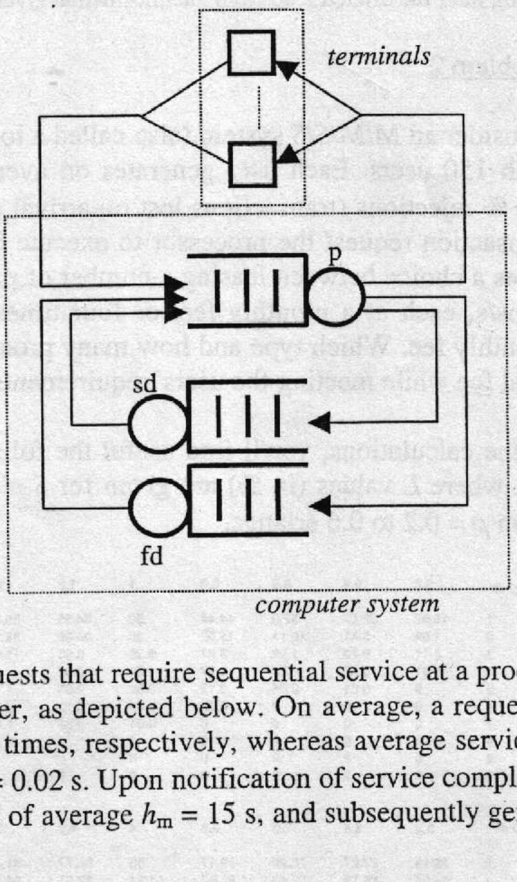
Operational Research / Queuing Systems
Practice Set 4

Problem 1

A single-processor queuing system interacts with J =10 intelligent terminals in a query-response manner, as depicted below. Having received a response, a terminal generates a new query after a think time of average duration h_m = 4 s. The average number of elementary operations needed to generate a response is b_m = 15000, and processor is v = 5000 elementary operations/s. Find the relationship between the proportion of processor idle time and mean *waiting* delay.



The flow conservation equation again comes in handy.



Problem 2

Each of J = 30 computer terminals generates requests that require sequential service at a processor, slow-disk controller and fast-disk controller, as depicted below. On average, a request has to visit these devices $l_p = 21$, $l_{sd} = 12$, $l_{fd} = 8$ times, respectively, whereas average service times there equal $\tau_p = 0.05$ s, $\tau_{sd} = 0.07$ s and $\tau_{fd} = 0.02$ s. Upon notification of service completion for its request, a terminal enters a think time of average h_m = 15 s, and subsequently generates another request.

(a) Which device is the bottleneck, and which one is the most overdimensioned? How will his change if the processor is tuned up so that $\tau_p = 0.03$ s?

(b) What processor speedup do we need in order for the mean system delay (request time within the system) to become d^*_m = 12 s, and what speedup would ensure d^*_m = 9 s?

Assume a certain request arrival interval at the terminal-system interface. Use it to express the offered load at each device. For (b), will Little's theorem be of any use?

Operational Research / Queuing Systems
Practice Set 5

Problem 1

Answer the following questions related to finite-buffer queuing systems.

(a) An average of 40 requests per second arrive in an M/M/1/2 queuing system, each requesting 20 ms of processor time. How many requests on average are lost per day due to buffer overflow?
(b) In an M/M/1/5 queuing system, two requests arrive on average during the service of a request of average length. What is the loss fraction?
(c) An M/M/1/Q queuing system serves an arrival stream whose mean characteristics are a_{m} and b_{m}. For sustained operation, the processor needs a $p\%$ proportion of idle time (used for maintenance), which in turn requires a minimum processor speed $v_{\min}$. How does $v_{\min}$ change with increasing Q?

In (b), infer the offered load from the information given.

Problem 2

Consider an M/M/S/S system (also called a loss system since it has no waiting room in buffer) with 150 users. Each user generates on average 10 transactions per second and tolerates $L \leq 3$ % rejections (transactions lost on arrival due to lack of available processors). An average transaction request the processor to execute 800 elementary operations. The system operator faces a choice between leasing a number of processors of speed 0.5 million elementary operations/s, each at a monthly fee, or four times faster processors leased at a 2.5 times higher monthly fee. Which type and how many processors should the operator lease to minimize the total fee while meeting the users' requirements?

In the calculations, you'll find useful the following table obtained from an Erlang-B calculator, where L values (in %) are given for S = 1 to 10 processors, and busy-hour load ranging from $\rho = 0.2$ to 0.6 erlangs.

$\rho=$	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2	2.2	2.4	2.6	2.8	3
S=															
1	16.67	28.57	37.5	44.44	50	54.55	58.33	61.54	64.29	66.67	68.75	70.59	72.22	73.68	75
2	1.64	5.41	10.11	15.09	20	24.66	28.99	32.99	36.65	40	43.06	45.86	48.42	50.78	52.94
3	0.11	0.72	1.98	3.87	6.25	8.98	11.92	14.96	18.03	21.05	24	26.84	29.56	32.15	34.61
4	0.01	0.07	0.3	0.77	1.54	2.62	4	5.65	7.5	9.52	11.66	13.87	16.12	18.37	20.61
5	0	0.01	0.04	0.12	0.31	0.63	1.11	1.77	2.63	3.67	4.88	6.24	7.73	9.33	11
6	0	0	0	0.02	0.05	0.12	0.26	0.47	0.78	1.21	1.76	2.44	3.24	4.17	5.21
7	0	0	0	0	0.01	0.02	0.05	0.11	0.2	0.34	0.55	0.83	1.19	1.64	2.19
8	0	0	0	0	0	0	0.01	0.02	0.05	0.09	0.15	0.25	0.39	0.57	0.81
9	0	0	0	0	0	0	0	0	0.01	0.02	0.04	0.07	0.11	0.18	0.27
10	0	0	0	0	0	0	0	0	0	0	0.01	0.02	0.03	0.05	0.08

$\rho=$	3.2	3.4	3.6	3.8	4	4.2	4.4	4.6	4.8	5	5.2	5.4	5.6	5.8	6
S=															
1	76.19	77.27	78.26	79.17	80	80.77	81.48	82.14	82.76	83.33	83.87	84.37	84.85	85.29	85.71
2	54.94	56.78	58.48	60.07	61.54	62.91	64.19	65.39	66.51	67.57	68.56	69.49	70.38	71.21	72
3	36.95	39.15	41.24	43.21	45.07	46.83	48.49	50.06	51.55	52.97	54.3	55.57	56.78	57.92	59.02
4	22.81	24.97	27.07	29.1	31.07	32.96	34.78	36.54	38.22	39.83	41.38	42.86	44.29	45.65	46.96
5	12.74	14.51	16.31	18.11	19.91	21.68	23.44	25.16	26.84	28.49	30.09	31.64	33.15	34.62	36.04
6	6.36	7.6	8.91	10.29	11.71	13.18	14.66	16.17	17.68	19.18	20.68	22.16	23.63	25.07	26.49
7	2.83	3.56	4.38	5.29	6.27	7.33	8.44	9.6	10.81	12.05	13.32	14.6	15.9	17.2	18.5
8	1.12	1.49	1.93	2.45	3.04	3.7	4.44	5.23	6.09	7	7.97	8.97	10.01	11.09	12.19
9	0.4	0.56	0.77	1.02	1.33	1.7	2.12	2.6	3.15	3.74	4.4	5.11	5.86	6.67	7.51
10	0.13	0.19	0.28	0.39	0.53	0.71	0.93	1.18	1.49	1.84	2.24	2.68	3.18	3.72	4.31

What is the maximum tolerable busy-hour load in either of the two options?

Problem 3

The arrival stream to an M/M/1 queuing system has mean interarrival interval $a_m = 10$ s and mean request size $b_m = 10$ s.u. Processor speed is v s.u./s. Draw a state transition diagram corresponding to the underlying birth-and-death process for the following model specifications:

a) with probability 25% a request whose service has been completed immediately returns to the queue instead of departing from the system,
b) at three or more requests in system, the processor speeds up by 50%,
c) upon termination of a busy period, the processor "goes on vacation" i.e., ignores arriving requests, and only resumes operation when three requests are queued,
d) the processor "goes on vacation" after completion of each request's service; "vacation" duration is exponentially distributed with mean h_m,
e) the processor occasionally breaks down and comes up after a while, whereupon the interrupted service is resumed (all requests arriving during breakdown are queued); the down- and up-times are exponentially distributed with mean f_m i g_m, respectively,
f) requests are admitted in pairs – the first request from a pair is held back until the second one arrives, then the pair is regarded as an arriving request of size equal to the size of the second request.

For each of these models carefully define a system state. Examine all events that can any moment occur at a given state, and the resulting state transitions.

Problem 3

Consider impatient requests in an M/M/1 system. The arrival stream has mean interval a_m = 10 s and mean request size b_m= 10 s.u., and processor speed is $v = 1$ s.u./s. On arrival, each request draws its individual "patience threshold" from exponential distribution with mean c_m, and upon its expiration escapes from the queue if still waiting, otherwise departs upon service completion. For ease of calculation let $c_m = b_m/v$. Find the distribution (p_k) of the number of requests in system, as well as the fraction L of escaped requests.

Use the general birth-and-death solution, also apply the flow conservation equation.