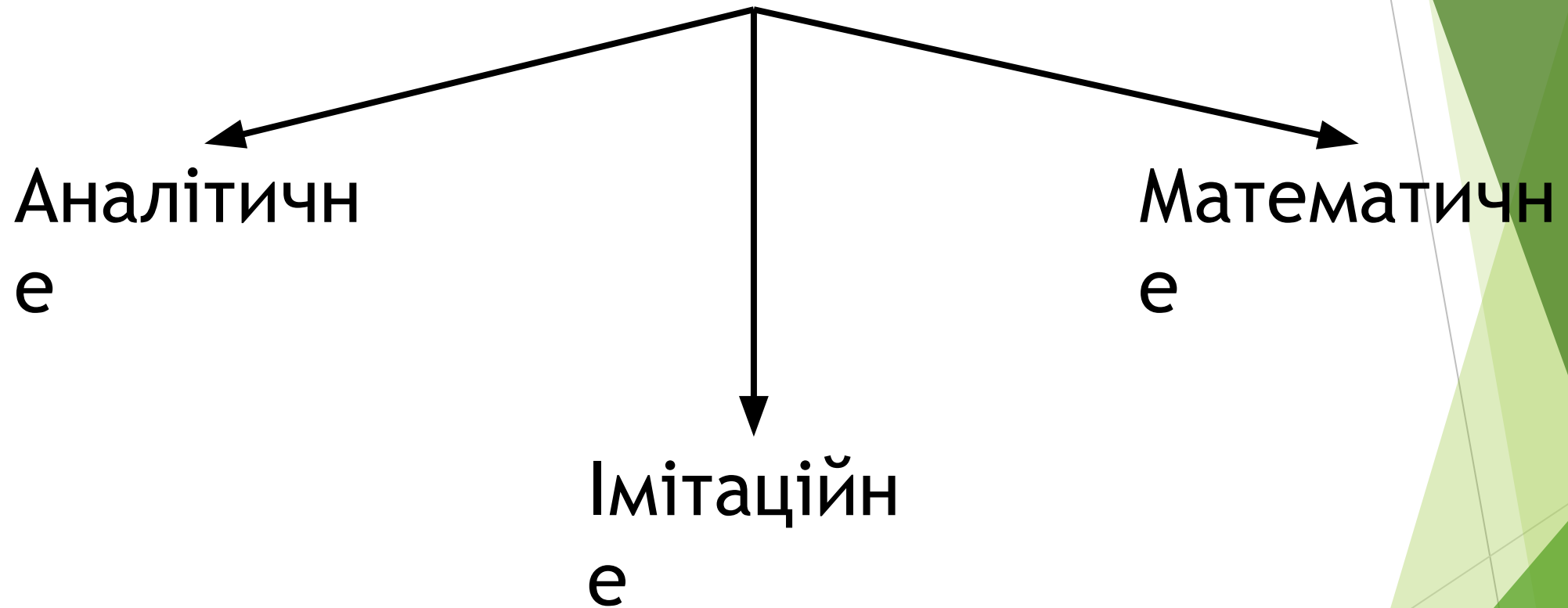
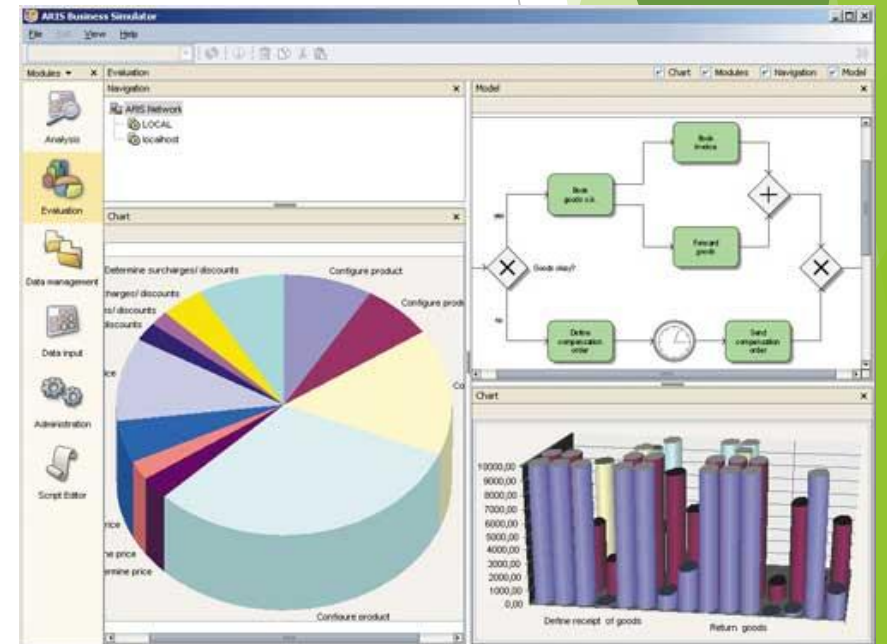
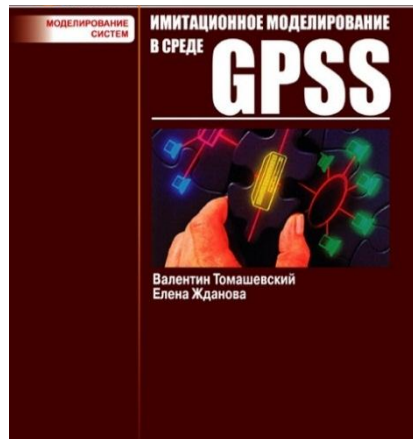
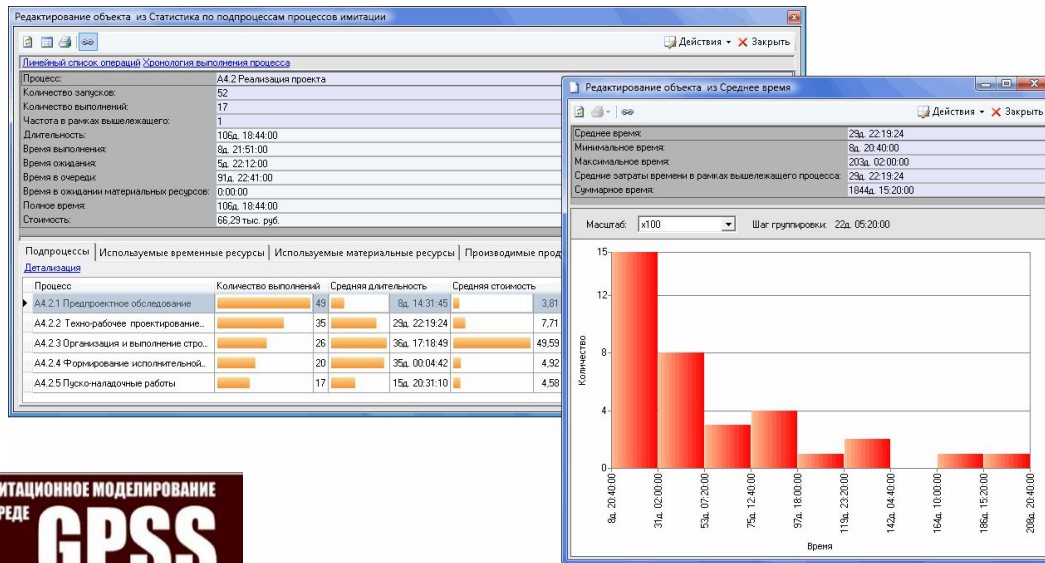


Методи моделювання



Імітаційне моделювання системи передбачає, що процес функціонування системи відтворюється за допомогою алгоритму, який реалізується за допомогою комп'ютера.



Генератори випадкових величин

- ▶ Імітаційні моделі складних систем містять випадкові величини, що мають різні закони розподілу. При побудові алгоритму імітації ці випадкові величини реалізуються генераторами випадкових чисел. Від якості генераторів випадкових чисел, що використовуються, залежить точність результатів імітаційного моделювання.
- ▶ **Генератор випадкових чисел** — обчислювальний або фізичний пристрій, спроектований для генерації послідовності номерів чи символів, які не відповідають будь-якому шаблону, тобто є випадковими. Широко використовуються комп'ютерні системи для генерації випадкових чисел, але часто вони малоефективні. Ці функції, можливо, забезпечують достатньо випадковості для певних завдань (наприклад, для відеоігор), але є непридатними в тих випадках, коли потрібна «високоякісна випадковість», як, наприклад, у криптографічних програмах, статистиці або чисельному аналізі.

экскурсия в расчетный центр

а здесь вы
можете видеть
наш генератор
случайных чисел



девять девять
девять девять
девять девять



вы
уверены
что эти
числа
случайны?



в этом вся
проблема со
случайностью:
ты никогда не
можешь быть
уверен



pikabu.ru

Способи генерування випадкових величин

- ▶ Зберігання у комп'ютері таблиці випадкових чисел і отримання потім з неї даних для імітаційного моделювання;
- ▶ **Недолік:** зберігання великого обсягу інформації та повільна швидкість
- ▶ Використання деякого фізичного пристрою, наприклад електронної лампи, для генерації випадкового шуму;
- ▶ **Недолік:** неможливість наведеного експерименту з параметрами моделі.
- ▶ Застосування рекурсивних формул коли на підставі i -того випадкового числа обчислюється $i+1$ -ше випадкове число
- ▶ Не має недоліків попередніх способів і в теперішній час є найбільш прийнятним

Вимоги до генераторів випадкових чисел

- ✓ Числа рівномірно розподілені на інтервалі $(0;1)$ і незалежні;
- ✓ Генерується достатньо велика кількість чисел, що не повторюються;
- ✓ Послідовність випадкових чисел може бути відтворена;
- ✓ Швидкодія;
- ✓ Обсяг пам'яті, що використовується, достатньо малий.

Якість генератора випадкових чисел перевіряють за допомогою машинного експерименту. Розрізняють тести двох типів - емпіричні та теоретичні.

Для генерування випадкового числа r , розподіленого за заданим законом $F(x)$, використовують такі методи

- ▶ метод оберненої функції;
- ▶ табличний метод;
- ▶ метод, оснований на функціональних властивостях законів розподілу.

ГЕНЕРУВАННЯ ВИПАДКОВОЇ ВЕЛИЧИНИ

Генерування рівномірно розподіленої в інтервалі (0;1)
випадкової величини ζ

Генерування випадкової величини r за заданим законом
розподілу $F(x)$

метод оберненої функції

$$r = F^{-1}(\zeta)$$

експоненціальний закон
розподілу

$$r = -\frac{1}{\lambda} \cdot \ln \zeta$$

табличний метод

$$r = x_{i-1} + \frac{x_i - x_{i-1}}{a_i - a_{i-1}} (\zeta - a_{i-1})$$

$$a_i = F(x_i)$$

емпіричний закон розподілу

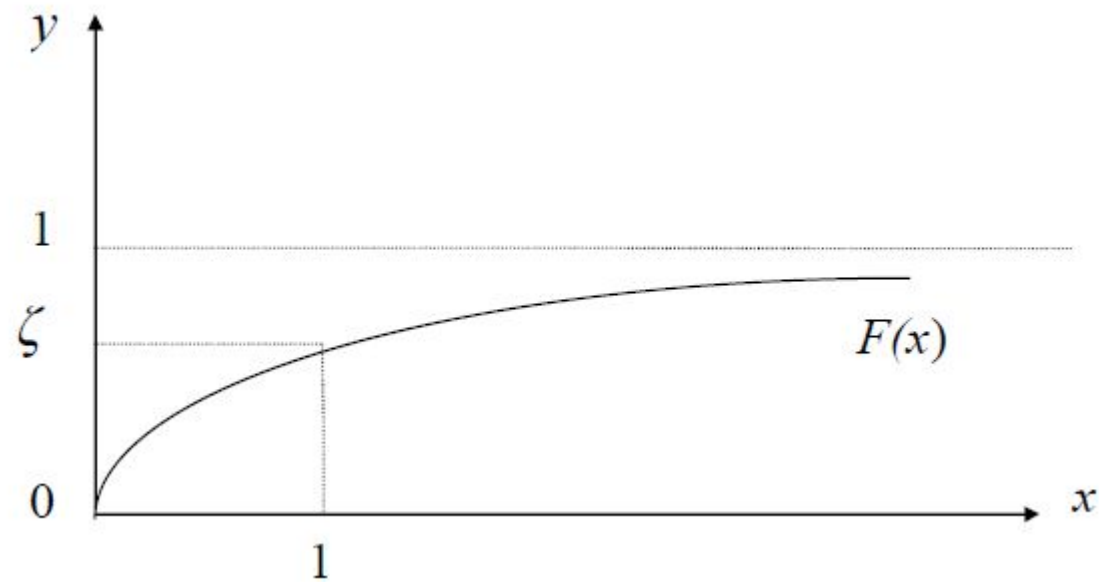
Спеціальні методи генерування

нормальний закон розподілу(закон Гауса) $r = \sigma \cdot \left(\sum_{i=1}^{12} \zeta_i - 6 \right) + a$

закон розподілу Ерланга

$$r = -\frac{1}{k\mu} \ln \left(\prod_{i=1}^k \zeta_i \right)$$

Генерування випадкової величини методом оберненої функції



Приклад 1 (Метод оберненої функції)

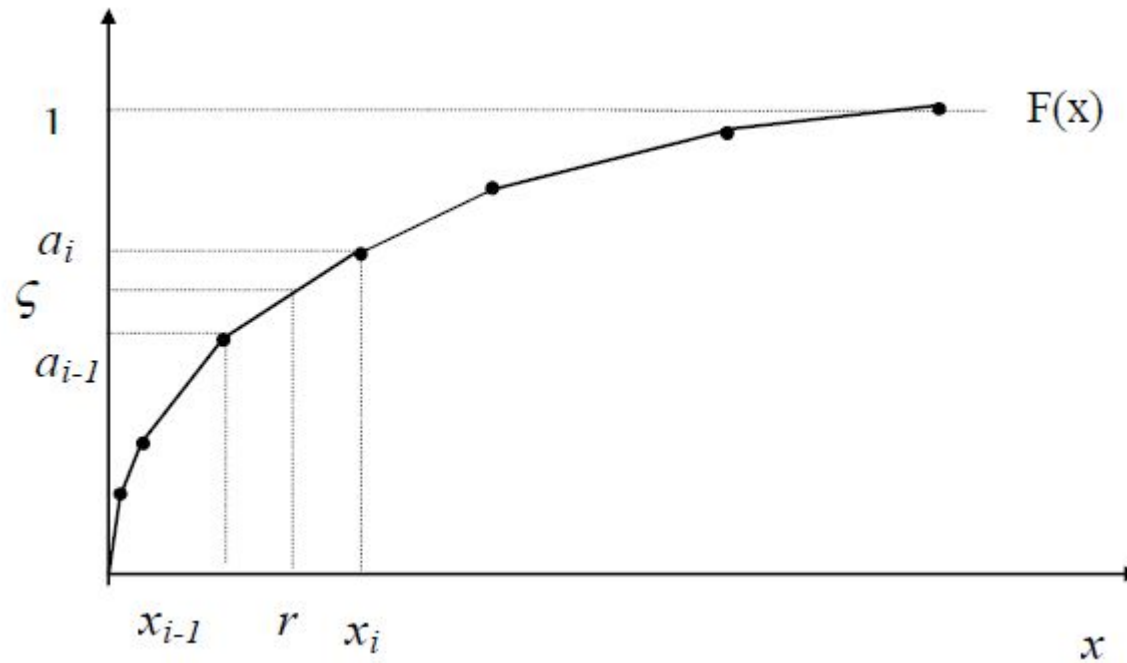
Для того, щоб отримати випадкову величину, яка розподілена за експоненціальним законом із параметром λ , потрібно розв'язати рівняння $\zeta = F(r)$:

$$\zeta = 1 - e^{-\lambda r} \Leftrightarrow r = -\frac{1}{\lambda} \cdot \ln(1 - \zeta).$$

Величина $1 - \zeta$ є випадкова величина, рівномірно розподілена на $(0,1)$. Тому цю дію робити не слід. Отже, маємо таку формулу для генерування випадкової величини, що має експоненціальний закон розподілу з параметром λ :

$$r = -\frac{1}{\lambda} \cdot \ln \zeta .$$

Табличний метод генерування випадкового числа r , що має закон розподілу $F(x)$



Приклад 2 (Табличний метод)

Приклад

Табличний метод використовується для генерування випадкових чисел, що мають емпіричний закон розподілу.

Припустимо, що емпіричний закон розподілу задано парами чисел (x_i, y_i) , $i = 1, \dots, n$. З'єднаємо дані пари чисел прямими лініями і отримаємо кусочно-лінійне наближення істинної функції закону розподілу. Для по-

будованого таким чином наближення функції закону розподілу використовують табличний метод генерування випадкових чисел

Методи, які використовують спеціальні властивості законів розподілу. Наприклад, для генерування нормально розподілених чисел використовується *метод полярних координат* Марсальї і Брея [Кельтон, Лоу]. За цим методом для того, щоб отримати нормально розподілену випадкову величини з математичним сподіванням 0 і дисперсією 1 потрібно виконати наступні кроки:

1) згенерувати дві рівномірно розподілені в інтервалі $(0;1)$ випадкові величини ζ_1, ζ_2 ,

2) перетворити величини ζ_1, ζ_2 у величини α_1, α_2 , які розподілені рівномірно в інтервалі $(-1; 1)$, за допомогою лінійного перетворення:

$$\alpha_i = 2\zeta_i - 1, \quad i = 1, 2;$$

3) якщо $\alpha_1^2 + \alpha_2^2 > 1$, то знову генерувати величини α_1, α_2 ; якщо величина $\alpha_1^2 + \alpha_2^2 \leq 1$, то сформувати величину

$$\beta = \sqrt{\frac{-2 \ln(\alpha_1^2 + \alpha_2^2)}{\alpha_1^2 + \alpha_2^2}};$$

4) сформувати величини

$$\gamma_i = \alpha_i \cdot \beta, \quad i = 1, 2.$$

Величини γ_1, γ_2 є незалежними нормально розподіленими випадковими величинами із математичним сподіванням 0 і дисперсією 1.

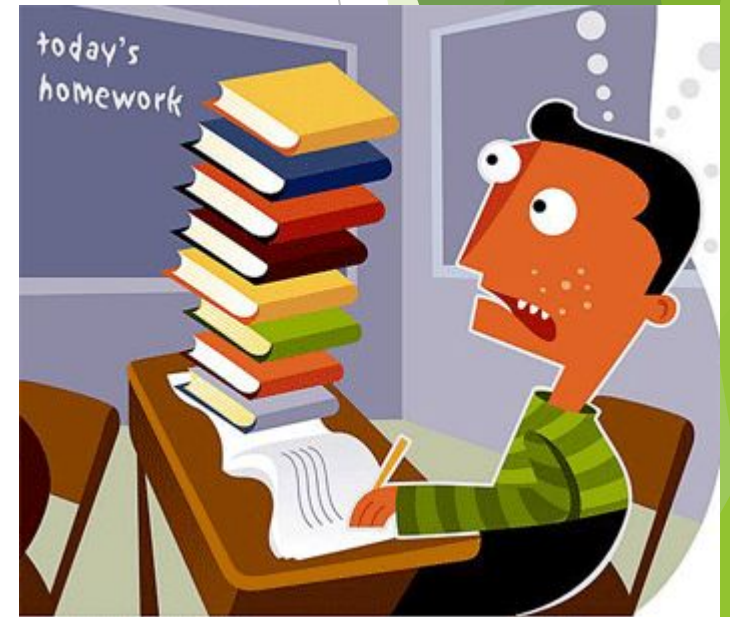
Приклади задач

Задача 1. Побудувати генератор випадкової величини, що має експоненціальний закон розподілу з параметром 2.

Розв'язання. Скористаємось методом оберненої функції. За формулою (5.12) отримуємо:

$$r = -\frac{1}{2} \cdot \ln \zeta,$$

де ζ є випадкова величина, рівномірно розподілена на $(0;1)$.



Відповідь: $r = -\frac{1}{2} \cdot \ln \zeta$, де ζ є випадкова величина, рівномірно розподілена на $(0;1)$.

Задача 2. Побудувати генератор випадкової величини r , що приймає значення 0, 1, 2 з ймовірностями 0,25, 0,5 та 0,25 відповідно.

Розв'язання. За формулою (5.13) для генерування дискретної випадкової величини отримуємо:

$$r = \begin{cases} 0, & \text{якщо } 0 < \zeta \leq 0,25 \\ 1, & \text{якщо } 0,25 < \zeta \leq 0,75 \\ 2, & \text{якщо } 0,75 < \zeta \leq 1 \end{cases}$$

де ζ є випадкова величина, рівномірно розподілена на $(0;1)$.

Відповідь: $r = \begin{cases} 0, & \text{якщо } 0 < \zeta \leq 0,25 \\ 1, & \text{якщо } 0,25 < \zeta \leq 0,75 \\ 2, & \text{якщо } 0,75 < \zeta \leq 1 \end{cases}$, де ζ є випадкова величина, рівномірно розподілена на $(0;1)$.

вномірно розподілена на $(0;1)$.

Задача 4. Побудувати генератор випадкових чисел, нормально розподілених з математичним сподіванням 10 та середнім квадратичним відхиленням 2.

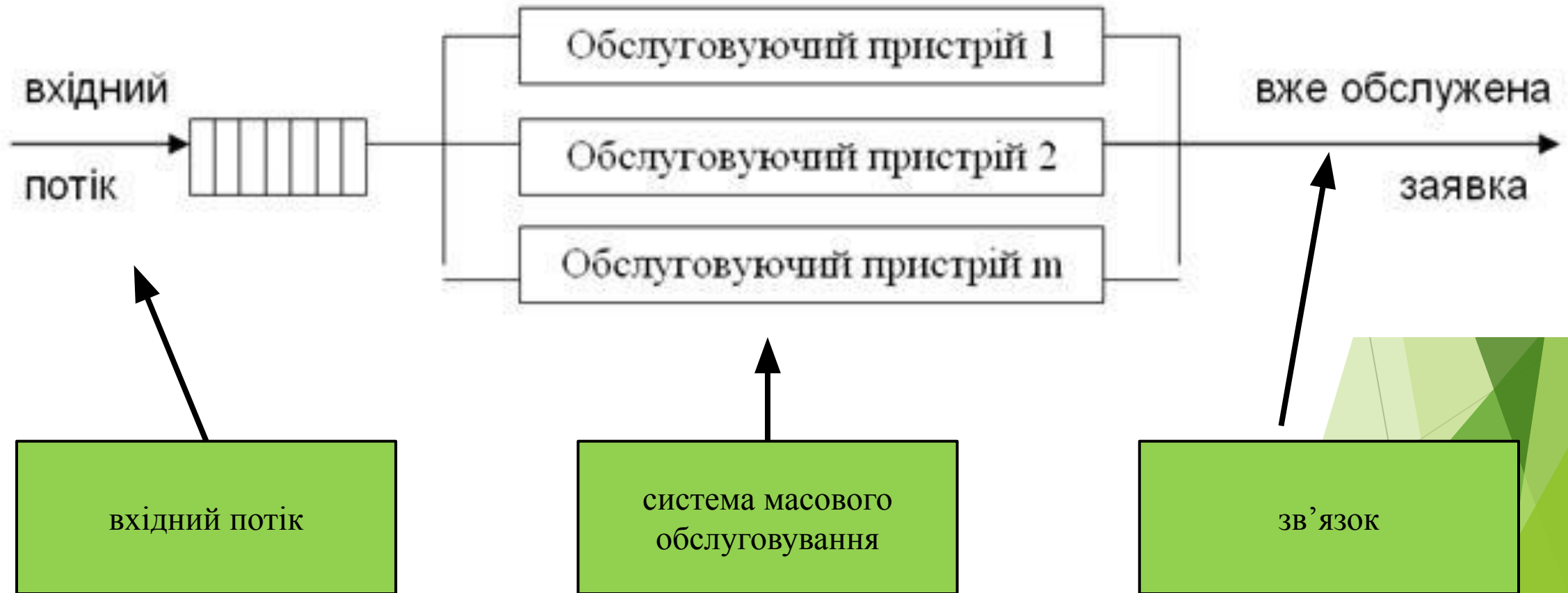
Розв'язання. Скористаємось формулами (5.16), (5.19). За умовами задачі математичне сподівання $a=10$, а середнє квадратичне відхилення $\sigma=2$. Тому маємо таку формулу

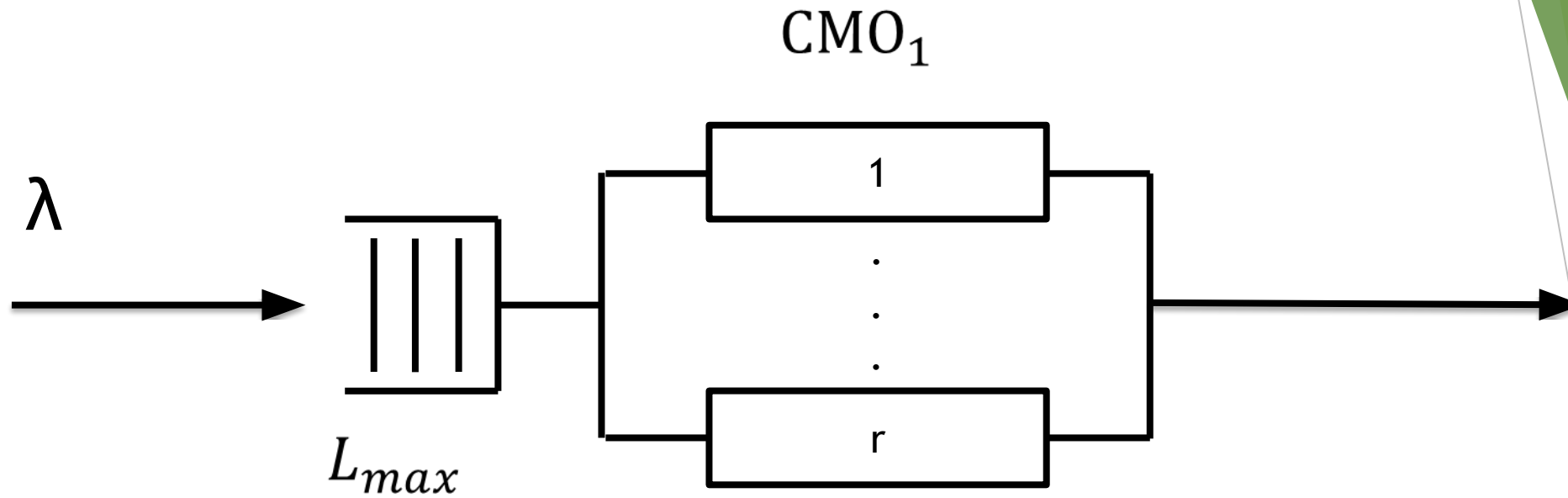
$$r = 2\gamma + 10,$$

величина γ у розрахована за формулами (5.16), (5.18).

Відповідь: генератор нормально розподілених випадкових чисел з математичним сподіванням 10 та середнім квадратичним відхиленням 2 задається формулою $r = 2\gamma + 10$.

Імітаційне моделювання ММО





r_i - кількість пристроїв в СМО

μ_i - інтенсивність обслуговування вимоги

r_i - кількість пристроїв в СМО

Num:=i, де $i = \begin{cases} 0, & \text{коли пристрій вільний} \\ 1, & \text{коли пристрій зайнятий} \\ 2, & \text{коли пристрій заблокований} \end{cases}$

Об'єктно-орієнтований підхід

Об'єкти, з яких складається мережа масового обслуговування:

- об'єкт «вхідний потік»
- об'єкт «СМО»
- об'єкт «маршрут»
- об'єкт «маршрут входу»
- об'єкт «маршрут виходу»

Задача

Служба замовлення таксі має 5 каналів для одночасного прийняття замовлень по телефону. Час між спробами виклику таксі розподілений за законом Ерланга другого порядку із середнім 180 секунд. Абонент витрачає 30 секунд для набирання номера і, якщо застає всі канали служби замовлення зайнятими або після з'єднання з'ясовує, що черга на обслуговування перевищує 10 замовлень (в такому випадку замовлення не приймаються), то через 60 секунд він повторює набирання номера. Після п'яти спроб абонент припиняє набирання. Служба замовлення таксі має у своєму розпорядженні 30 машин таксі для обслуговування замовлень. Час, витрачений на проїзд до клієнта, залежить від відстані до нього. Ймовірності можливих відстаней розподіляються таким чином:

Відстань, км	5	8	9	11	12	20
Ймовірність	0,1	0,2	0,25	0,17	0,23	0,05

Вартість проїзду до клієнта клієнтом не сплачується. Швидкість руху машин рівномірно розподілена в інтервалі 45 ± 5 км/год. Час обслуговування клієнта рівномірно розподілений в інтервалі 50 ± 20 хвилин. Вартість попереднього замовлення складає 2 гривні, вартість проїзду 1 км складає 2 гривні².

Служба замовлення таксі має 5 каналів для одночасного прийняття замовлень по телефону.

Час між спробами виклику таксі розподілений за законом Ерланга другого порядку із середнім 180 секунд.

Абонент затрачає 30 секунд для набирання номера.

Зайняття телефоніста (одного з п'яти) моделюється наступним фрагментом:

TEL	STORAGE	5
	ENTER	TEL
	ADVANCE	30
	LEAVE	TEL

Якщо застає всі канали служби замовлення зайнятими то через 60 секунд він повторює набирання номера.

Після п'яти спроб абонент припиняє набирання.

**TEL
DOZVON**

**BOTH,VIDP,VIDM
VIDP**

VIDM

**STORAGE 5
ADVANCE 30
TRANSFER**

**ENTER TEL
ADVANCE 30
LEAVE TEL
ADVANCE 60
LOOP 1,DOZVON
TERMINATE**

Клієнт відмовляється від обслуговування також, якщо дізнався, що черга клієнтів на машини таксі надто велика (більша 10).

TEL

**STORAGE 5
DOZVON ADVANCE 30
TRANSFER**

**BOTH,VIDP,VIDM
VIDP**

**ENTER TEL
ADVANCE 30
LEAVE TEL
TEST L**

Q\$KLIENT,10,VIDM

SAVEVALUE

**DOHOD+,2000.
VIDM**

**ADVANCE 60
LOOP 1,DOZVON
TERMINATE**

Служба замовлення таксі має у своєму розпорядженні 30 машин таксі для обслуговування замовлень. Час, витрачений на проїзд до клієнта, залежить від відстані до нього.

Вартість проїзду до клієнта клієнтом не сплачується. Швидкість руху машин рівномірно розподілена в інтервалі 45 ± 5 км/год. Час обслуговування клієнта рівномірно розподілений в інтервалі 50 ± 20 хвилин. Вартість попереднього замовлення складає 2 гривні, вартість проїзду 1 км складає 2 гривні².

DOBSL	FUNCTION RN2,C2 0.,1800/.9999,4200
DSPEED	FUNCTION RN3,C2 0.,11.111/.9999,13.888
VIDST	FUNCTION RN4,D6 0.1,5000/0.3,8000/0.55,9000/0.72,11000/0.9 5,12000/1.00,20000
CHAS	VARIABLE
FN\$VIDST/FN\$DSPEED	
COST	VARIABLE
FN\$DOBSL#FN\$DSPEED#0.2	
OBSL	QUEUE KLIENT ENTER TAXI DEPART KLIENT ADVANCE V\$CHAS ADVANCE FN\$DOBSL LEAVE TAXI SAVEVALUE



```

TAXI      STORAGE  30          ;кількість машин таксі
TEL       STORAGE  5          ;кількість телефоністів
DIS       FUNCTION  RN1,C24    ;функція експоненціального закону розподілу
0.,0./ .100,.104/.200,.222/.3,.355/.4,.509/.5,.69/.6,.915/.7,1.2/.750,1.38
.8,1.6/.84,1.83/.88,2.12/.9,2.3/.92,2.52/.94,2.81/.95,2.99/.96,3.2
.97,3.5/.98,3.9/.99,4.6/.995,5.3/.998,6.2/.999,7/.9998,8
DOBSL     FUNCTION  RN2,C2     ;функція час обслуговування клієнта в секундах
0.,1800/.9999,4200
DSPEED    FUNCTION  RN3,C2
0.,11.111/.9999,13.888          ;функція швидкості руху машини в метрах на секунду
VIDST     FUNCTION  RN4,D6     ;функція, що задає відстань до клієнта в метрах
0.1,5000/0.3,8000/0.55,9000/0.72,11000/0.95,12000/1.00,20000
CHAS      VARIABLE  FN$VIDST/FN$DSPEED ;час руху машини до клієнта
COST      VARIABLE  FN$DOBSL#FN$DSPEED#0.2 ;вартість обслуговування в копійках за метр
          GENERATE  90, FN$DIS   ;з'явився клієнт, що бажає таксі
          ADVANCE   90, FN$DIS   ;затримка ерлангового розподілу
          ASSIGN   1,5          ;не більше п'яти спроб набирання номера
DOZVON    ADVANCE   30          ;набирання номера
          TRANSFER  BOTH,VIDP,VIDM ;якщо телефоніст вільний то відповідь інакше відмова
VIDP      ENTER    TEL          ;зайняти телефоніста
          ADVANCE   30          ;тривалість розмови в секундах
          LEAVE    TEL          ;звільнити телефоніста
          TEST L   Q$KLIENT,10,VIDM ;якщо черга на машини більше 10, то відмова
          SAVEVALUE DOHOD+,200   ;підрахувати дохід за попереднє замовлення
          TRANSFER ,OBSL        ;перейти до обслуговування
VIDM      ADVANCE   60          ;затримка після невдалого дзвінка
          LOOP     1,DOZVON     ;цикл, що здійснює дозвон
          SAVEVALUE NEOBSL+,1   ;підрахувати необслугованих клієнтів
          TERMINATE ;вихід із системи клієнта, що не дозвонився
OBSL      QUEUE    KLIENT      ;очікувати вільне таксі
          ENTER    TAXI         ;таксі виконує замовлення
          DEPART   KLIENT      ;зменшити кількість очікуючих клієнтів на одиницю
          ADVANCE  V$CHAS       ;таксі прямує до клієнта
          ADVANCE  FN$DOBSL     ;таксі обслуговує клієнта
          LEAVE    TAXI         ;звільнити таксі
          SAVEVALUE DOHOD+,V$COST ;підрахувати дохід за обслуговування
          SAVEVALUE KLOBS+,1    ;підрахувати кількість обслугованих клієнтів
          TERMINATE ;вихід із системи клієнта, що обслуговувався
          GENERATE 864000       ;час моделювання 240 годин
          TERMINATE 1

```

LABEL	LOC	BLOCK TYPE	ENTRY COUNT	CURRENT COUNT	RETRY
	1	GENERATE	9627	0	0
	2	ADVANCE	9627	4	0
	3	ASSIGN	9623	0	0
DOZVON	4	ADVANCE	28823	0	0
	5	TRANSFER	28823	0	0
VIDP	6	ENTER	28487	0	0
	7	ADVANCE	28487	1	0
	8	LEAVE	28486	0	0
	9	TEST	28486	0	0
	10	SAVEVALUE	6833	0	0
	11	TRANSFER	6833	0	0
VIDM	12	ADVANCE	21989	1	0
	13	LOOP	21988	0	0
	14	SAVEVALUE	2788	0	0
	15	TERMINATE	2788	0	0
OBSL	16	QUEUE	6833	10	0
	17	ENTER	6823	0	0
	18	DEPART	6823	0	0
	19	ADVANCE	6823	2	0
	20	ADVANCE	6821	28	0
	21	LEAVE	6793	0	0
	22	SAVEVALUE	6793	0	0
	23	SAVEVALUE	6793	0	0
	24	TERMINATE	6793	0	0
	25	GENERATE	1	0	0
	26	TERMINATE	1	0	0

QUEUE	MAX CONT.	ENTRY	ENTRY(0)	AVE.CONT.	AVE.TIME	AVE.(-0)	RETRY
KLIENT	10	10	6833	30	9.306	1176.647	1181.835 0

STORAGE	CAP.	REM.	MIN.	MAX.	ENTRIES	AVL.	AVE.C.	UTIL.	RETRY	DELAY
TAXI	30	0	0	30	6823	1	29.949	0.998	0	10
TEL	5	4	0	5	28487	1	0.989	0.198	0	0

SAVEVALUE	RETRY	VALUE
DOHOD	0	52311875.943
KLOBS	0	6793.000
NEOBSL	0	2788.000

Система імітаційного моделювання PTRSIM

Основною ідеєю системи є автоматизація імітаційного моделювання систем, представлених мережами Петрі.

Технологія імітаційного моделювання PTRSIM складається з таких етапів:

- 1) побудова формалізованої моделі засобами мереж Петрі;
- 2) введення моделі в імітаційну систему PTRSIM;
- 3) перевірка моделі на відповідність задуму моделювання;
- 4) виконання імітації при заданих значеннях параметрів;
- 5) виведення результатів імітації;
- 6) формулювання висновків і пропозицій.

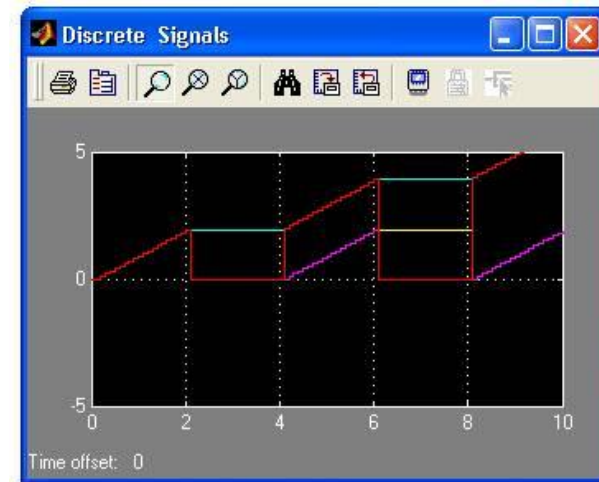
Пакет імітаційного моделювання Arena

Основа технології Arena - мова моделювання SIMAN і система Cinema Animation.

Пакет Arena містить програми, призначені для дослідженню властивостей системи Input Analyzer, Output Analyzer, Process Analyzer. Професійна версія пакету містить також програму, що підтримує пошук оптимальних значень параметрів системи.

Імітаційне моделювання

- ▶ Імітаційне моделювання – це окремий вид математичного моделювання. Імітаційна модель – це логіко-математичний опис об'єкта, який можна застосовувати для експериментування на комп'ютері з метою проектування, аналізу та оцінювання функціонування об'єкта.



Імітаційне моделювання мережі Петрі з часовими затримками

- ▶ Алгоритм імітації мережі Петрі складається з опису елементів моделі та опису правил зміни стану елементів.
- ▶ Елементами мережі Петрі являються:
 - ▶ § позиція,
 - ▶ § перехід,
 - ▶ § зв'язок
- ▶ Стан позиції повністю описується кількістю маркерів у позиції.

Умова запуску переходу T_j :

$$M_i \geq D_{ji}^- \text{ для усіх } i.$$

Запуск переходу T_j з часовою затримкою здійснюється в два етапи.

1) $M_i' = M_i - D_{ji}^-$ для усіх i та для заданого j .

2) $M_i'' = M_i' + D_{ji}^+$ для усіх i та для заданого j .

Поточний момент часу t встановлюється у вибране найменше зна-

▶ чення t_{min} і здійснюється запуск переходу:

$$t := t_{min},$$

$$M_i := M_i + D_{j_{min}i}^+ \text{ для усіх } i$$

Алгоритм імітації мережі Петрі (матриця)

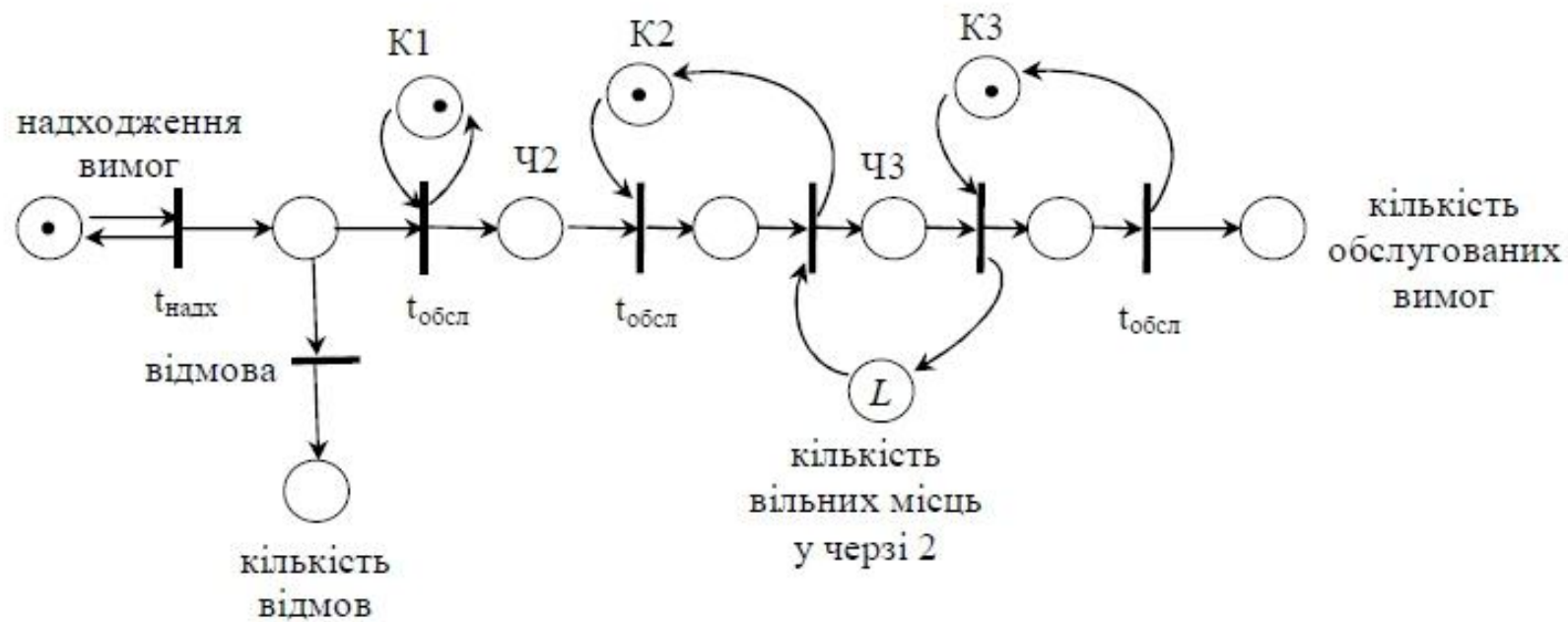


Рисунок 5.12. Мережа Петрі з часовими затримками, що моделює мережу масового обслуговування

Імітаційне моделювання мережі Петрі з конфліктними переходами

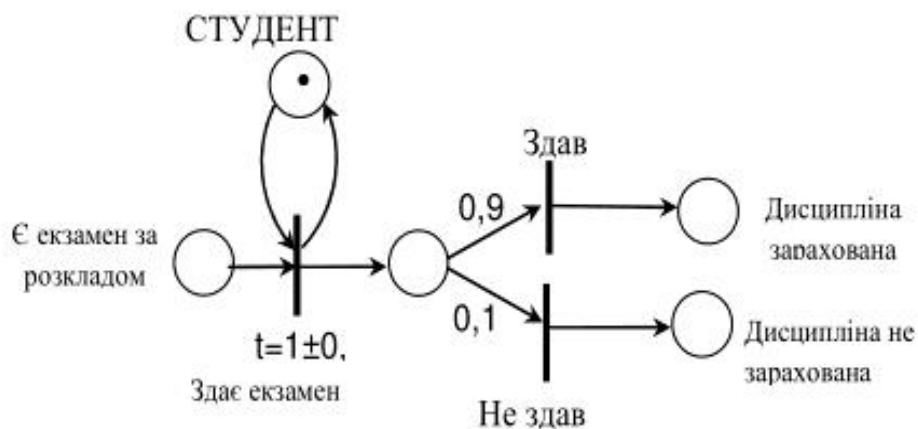


Рис. 1. Формалізація події Екзамен об'єкта Студент мережею Петрі

- ▶ Алгоритм імітації мережі Петрі з конфліктними переходами ускладнюється тим, що спочатку перевіряється умова запуску переходів, потім розв'язується конфлікт між переходами, для яких одночасно виконана умова запуску, і, нарешті, виконується запуск переходу.

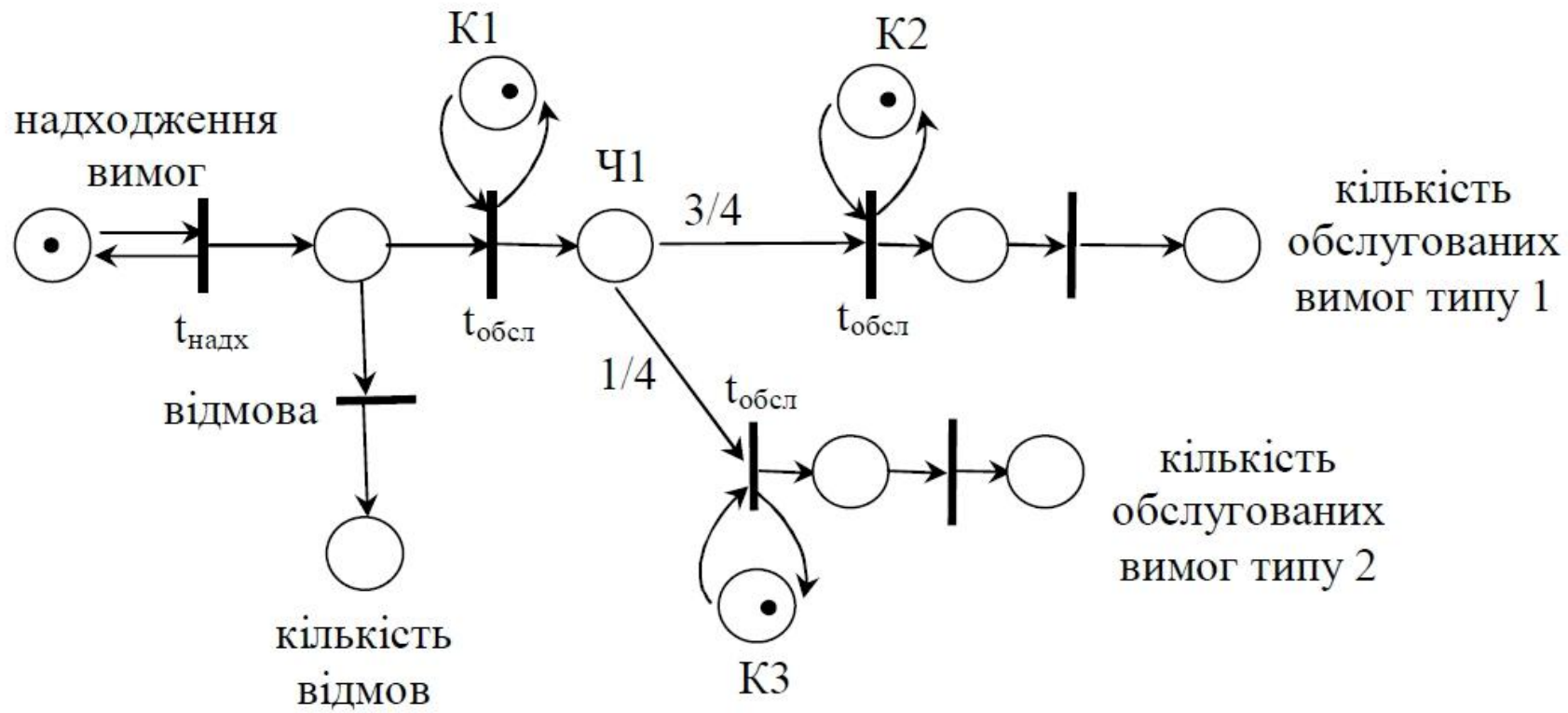
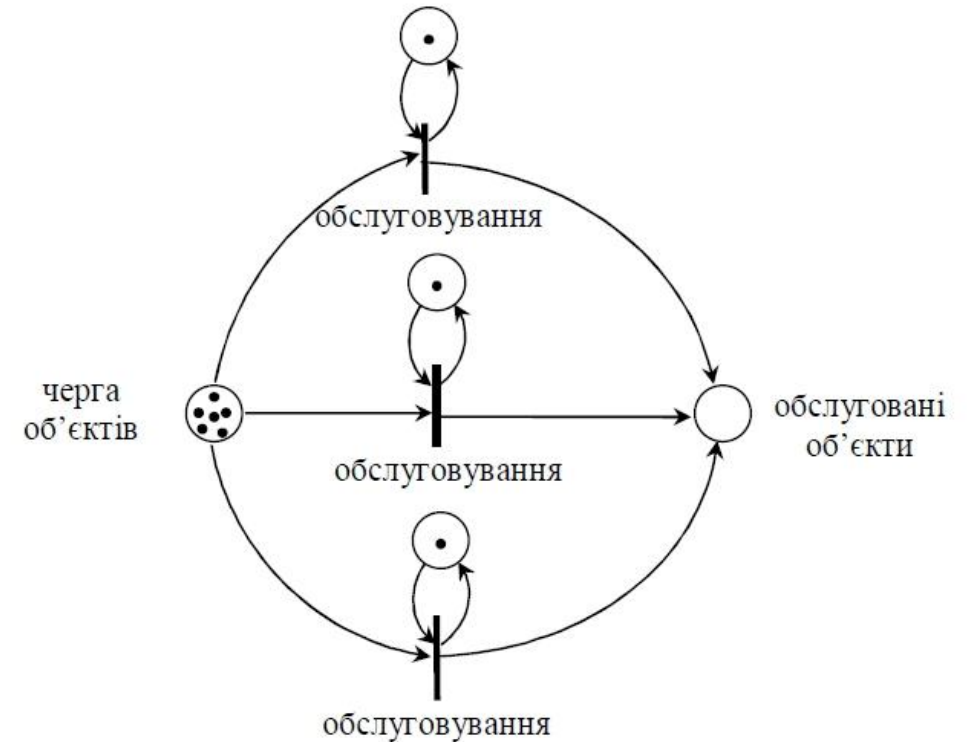
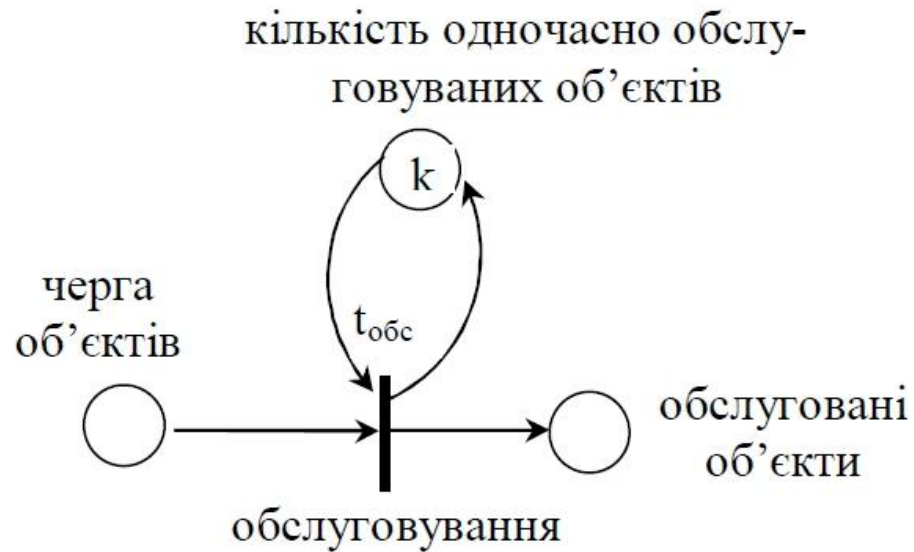


Рисунок 5.13. Мережа Петрі з часовими затримками та з конфліктними переходами

Імітаційне моделювання мережі Петрі з багатоканальними переходами



Імітаційне моделювання мережі Петрі з багатоканальними переходами

- ▶ Алгоритм імітації мережі Петрі з багатоканальними переходами ускладнюється, по-перше, тим, що замість одного значення t_{outj} у переході, що є багатоканальним, доведеться запам'ятовувати масив значень t_{outj} .
- ▶ По-друге, доведеться передбачити багаторазовий запуск переходу для чого перевірку умови запуску переходу та запуск переходів потрібно здійснювати доти, доки виконується умова запуску хоч для одного з них.

Імітаційне моделювання мережі Петрі з багатоканальними переходами

- ▶ Матричний підхід до побудови алгоритму імітації мережі Петрі ефективний тільки при відносно невеликій кількості переходів та позицій.
- ▶ При великій кількості переходів та позицій матриці входів та виходів містять, як правило, велику кількість нулів, перегляд яких при кожному просуванні модельного часу гальмує процес імітації.

- ▶ Об'єктно-орієнтований підхід дозволяє задати структурні зв'язки
- ▶ між елементами мережі Петрі за допомогою об'єктів типу зв'язок. Перевірка умови запуску переходу в цьому випадку здійснюється тільки для позицій, що являються для переходу вхідними. На основі об'єктно-орієнтованого підходу побудована універсальна система імітаційного моделювання PTRSIM