МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО

Навчально-науковий інститут електричної інженерії

та інформаційних технологій

КАФЕДРА АВТОМАТИЗАЦІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

ЗВІТ

З ЛАБОРАТОРНИХ РОБІТ

З НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«Моделювання систем»

Виконав студент групи КН-23-1

Іщенко Євген Володимирович

Перевірив доцент кафедри АІС Бурдільна Є. В.

КРЕМЕНЧУК 2025

**Лабораторна робота № 3**

**Тема:** Моделювання дискретних випадкових величин та потоків подій

**Мета:** навчитися розв’язувати задачі моделювання випадкових подій і випадкових величин із заданим законом розподілу з метою подальшого використання в задачах імітаційного моделювання.

**Виконання завдання лабораторної роботи:**

1. Побудувати процедуру імітації дискретної випадкової величини Х з законом розподілення, завданим у наступній таблиці 3.1.

Таблиця 3.1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Варіант |  |  |  |  |  |  |  |
| 5 | 0,25 | 0,5 | 0,15 | 0,05 | 0,05 |  |  |

Випадкова величина приймає значень з імовірностями

Тоді функцію розподілення можна визначити наступним чином:

*…………*

Для розв’язання поставленого завдання можна застосувати метод оберненої функції, тобто знайти випадкову величину за допомогою перетворення де , – функція, обернена до.

Простіший алгоритм обчислення дискретної випадкової величини , який заданий таблицею розподілення:

Якщо , то інакше,

Якщо , то інакше,

Якщо , то інакше

Геометрична інтерпретація алгоритму зведена до наступного: одиничний відрізок ділиться на n ділянок довжиною . Якщо випадкове число припало, наприклад, на ділянку , то це означає, що як значення випадкової величини потрібно вибрати .

using System;

class Program

{

static void Main()

{

Console.OutputEncoding = System.Text.Encoding.UTF8;

// Задаємо ймовірності для X\_1–X\_5

double[] probabilities = { 0.25, 0.5, 0.15, 0.05, 0.05 };

double[] cumulative = new double[probabilities.Length + 1];

cumulative[0] = 0;

// Розрахунок меж (точок) для проміжків

for (int i = 0; i < probabilities.Length; i++)

{

cumulative[i + 1] = Math.Round(cumulative[i] + probabilities[i], 2);

}

// Виводимо точки проміжків

Console.Write("Точки проміжків: ");

for (int i = 1; i < cumulative.Length; i++)

{

Console.Write($"{cumulative[i]}; ");

}

Console.WriteLine();

// Лічильники

int[] counts = new int[probabilities.Length];

// Моделювання

Random rnd = new Random();

for (int i = 0; i < 10000; i++)

{

double value = rnd.NextDouble();

for (int j = 0; j < probabilities.Length; j++)

{

if (value > cumulative[j] && value <= cumulative[j + 1])

{

counts[j]++;

break;

}

}

}

// Вивід результатів

for (int i = 0; i < probabilities.Length; i++)

{

Console.WriteLine($"Випадкове значення потрапило у проміжок {cumulative[i]} - {cumulative[i + 1]}: {counts[i]} разів");

}

}

}

Точки проміжків: 0.25; 0.75; 0.9; 0.95; 1;

Випадкове значення потрапило у проміжок 0 - 0.25: 2521 разів

Випадкове значення потрапило у проміжок 0.25 - 0.75: 4964 разів

Випадкове значення потрапило у проміжок 0.75 - 0.9: 1508 разів

Випадкове значення потрапило у проміжок 0.9 - 0.95: 513 разів

Випадкове значення потрапило у проміжок 0.95 - 1: 494 разів

1. Створити процедуру моделювання випадкової величини з експоненціальним законом розподілення з параметром , який заданий у таблиці 3.2.

Таблиця 3.2

|  |  |  |
| --- | --- | --- |
| Варіант | 5 |  |
|  | 1,5 | 0,5 |

Для моделювання випадкової величини з експоненціальним законом розподілення з параметром можна також використовувати обернену функцію. Експоненціальна щільність розподілення випадкової величини має вигляд:

оберненою функцією буде функція

де – набір псевдовипадкових чисел, які отримані, наприклад, за допомогою функції rand.

using System;

using System.Collections.Generic;

class Program

{

static void Main()

{

Console.OutputEncoding = System.Text.Encoding.UTF8;

double lambda = 1.5;

List<double> Xn = new List<double>();

Random rnd = new Random();

Console.WriteLine($"Генерація 10 випадкових чисел з експоненціальним розподілом при λ = {lambda}");

for (int i = 0; i < 10; i++)

{

double y = rnd.NextDouble(); // Випадкове число в [0,1)

// Застосування оберненої функції розподілу

double xn = -Math.Log(1 - y) / lambda;

Console.WriteLine($"X[{i + 1}] = {xn:F4}");

Xn.Add(xn);

}

}

}

Генерація 10 випадкових чисел з експоненціальним розподілом при λ = 1.5

X[1] = 1.0178

X[2] = 0.4316

X[3] = 0.6744

X[4] = 2.2162

X[5] = 0.5280

X[6] = 0.2787

X[7] = 0.7100

X[8] = 0.3095

X[9] = 0.6535

X[10] = 0.0860

1. За допомогою складеної процедури моделювання випадкової величини з експоненціальним законом розподілення змоделювати потік подій, у якому інтервали часу між подіями мають експоненціальне розподілення з параметром λ. Результати моделювання відобразити на осі часу з позначенням моментів виникнення подій.

Для моделювання потоку подій, у якому інтервали часу між подіями розподілені за довільним законом, можна скористатися наступним алгоритмом:

1. За допомогою генератора псевдовипадкових чисел і оберненої функції отримати ряд значень
2. Нанести їх на вісь часу наступним чином:
   * перша подія настає після одиниць часу після початку моделювання,
   * друга подія настає після одиниць часу після початку моделювання і так далі.

using System;

using System.Collections.Generic;

class Program

{

static void Main()

{

Console.OutputEncoding = System.Text.Encoding.UTF8;

double lambda = 1.5;

int numberOfEvents = 10; // Кількість подій для моделювання

List<double> interEventTimes = new List<double>(); // Список інтервалів між подіями

List<double> eventTimes = new List<double>(); // Час виникнення кожної події

Random rnd = new Random();

double currentTime = 0;

Console.WriteLine($"Моделювання потоку подій з експоненціальним розподілом (λ = {lambda})");

Console.WriteLine();

for (int i = 0; i < numberOfEvents; i++)

{

double y = rnd.NextDouble(); // Генерація псевдовипадкового числа в [0,1)

double interval = -Math.Log(1 - y) / lambda; // Обернена функція експоненційного розподілу

interEventTimes.Add(interval);

currentTime += interval;

eventTimes.Add(currentTime);

Console.WriteLine($"Подія {i + 1} настала через {interval:F4} одиниць часу, сумарний час: {currentTime:F4}");

}

Console.WriteLine("\nМоменти виникнення подій на осі часу:");

foreach (var time in eventTimes)

{

Console.WriteLine($"| {time:F4} ", time);

}

}

}

Моделювання потоку подій з експоненціальним розподілом (λ = 1.5)

Подія 1 настала через 1.2165 одиниць часу, сумарний час: 1.2165

Подія 2 настала через 0.0645 одиниць часу, сумарний час: 1.2809

Подія 3 настала через 0.1809 одиниць часу, сумарний час: 1.4618

Подія 4 настала через 0.0372 одиниць часу, сумарний час: 1.4990

Подія 5 настала через 1.5137 одиниць часу, сумарний час: 3.0127

Подія 6 настала через 1.2541 одиниць часу, сумарний час: 4.2668

Подія 7 настала через 0.0483 одиниць часу, сумарний час: 4.3150

Подія 8 настала через 0.1454 одиниць часу, сумарний час: 4.4605

Подія 9 настала через 0.0445 одиниць часу, сумарний час: 4.5049

Подія 10 настала через 0.5799 одиниць часу, сумарний час: 5.0849

**Висновок:** У ході лабораторної роботи було вивчено принципи моделювання випадкових процесів та розроблено програмні засоби для імітації дискретних випадкових величин і потоків подій. Основна увага приділялася роботі з експоненціальним розподілом, який широко використовується в теорії ймовірностей. Було створено спеціалізовані програми, що дозволяють генерувати випадкові величини з заданими параметрами та аналізувати їх характеристики. Всі параметри моделей підбиралися індивідуально згідно з варіантом завдання, що забезпечило глибоке розуміння принципів роботи з випадковими процесами. Отримані результати демонструють ефективність використаних методів і можуть бути застосовані для вирішення більш складних завдань імітаційного моделювання в майбутніх дослідженнях.

**Контрольні питання:**

1. **Що таке дискретна випадкова величина?**

Дискретна випадкова величина — це величина, яка може набувати лише окремих, ізольованих значень (як правило, цілих чисел), кожному з яких відповідає певна ймовірність. Прикладом дискретної випадкової величини може бути кількість влучень у мішень або кількість заявок, що надійшли за одиницю часу.

1. **Як задається закон розподілення дискретних випадкових величин?**

Закон розподілення дискретної випадкової величини задається у вигляді таблиці, в якій кожному можливому значенню випадкової величини відповідає певна ймовірність. Сума всіх ймовірностей повинна дорівнювати 1. Такий розподіл можна задати як таблично, так і графічно.

1. **Яким чином робиться моделювання дискретних випадкових величин?**

Моделювання дискретної випадкової величини здійснюється методом кумулятивної суми (накопиченої ймовірності). Генерується випадкове число з інтервалу [0; 1), після чого визначається, до якого з інтервалів накопичених ймовірностей воно належить. Значення випадкової величини, яке відповідає цьому інтервалу, і є результатом моделювання.

1. **Яким чином робиться моделювання випадкових величин з довільним законом розподілення?**

Для моделювання випадкових величин із довільним законом розподілення використовують метод оберненої функції розподілу. Полягає він у наступному:

1. генерується випадкове число Y з рівномірного розподілу на [0;1);
2. обчислюється значення X = F⁻¹(Y), де F⁻¹ — обернена функція до функції розподілу F(x).
3. Як можна змоделювати потік подій, у якому інтервали часу між подіями мають експоненціальне розподілення?

Для моделювання потоку подій з експоненціально розподіленими інтервалами часу застосовують обернений метод. Спочатку генерується послідовність випадкових величин X₁, X₂, ..., Xₙ, що мають експоненційний розподіл за формулою:

X = -ln(1 - Y) / λ, де Y ∈ [0;1)

Далі визначаються моменти часу настання подій як сума попередніх інтервалів:

T₁ = X₁

T₂ = X₁ + X₂

T₃ = X₁ + X₂ + X₃

...

Таким чином, формується часовий ряд, який моделює потік подій.