

Міністерство освіти і науки України
Національний технічний університет України «КПІ»
імені Ігоря Сікорського

ЗВІТ
з лабораторної роботи №3
з дисципліни «Мультипарадигмне програмування»

Виконав:
Студент 3 курсу кафедри ОТ ФІОТ,
Навчальної групи ІО-23
Прохоренко Артем

Київ 2025

Завдання: за допомогою мультипарадигмної мови R реалізувати перетворення чисельного ряду лінгвістичного ланцюжка за певним розподілом ймовірностей потрапляння значень до інтервалів з подальшою побудовою матриці передування.

Вхідні данні: чисельний ряд, вид розподілу ймовірностей, потужність алфавіту.

Вихідні дані: лінгвістичний ряд та матриця передування.

Мова програмування: R.

Варіант: 20

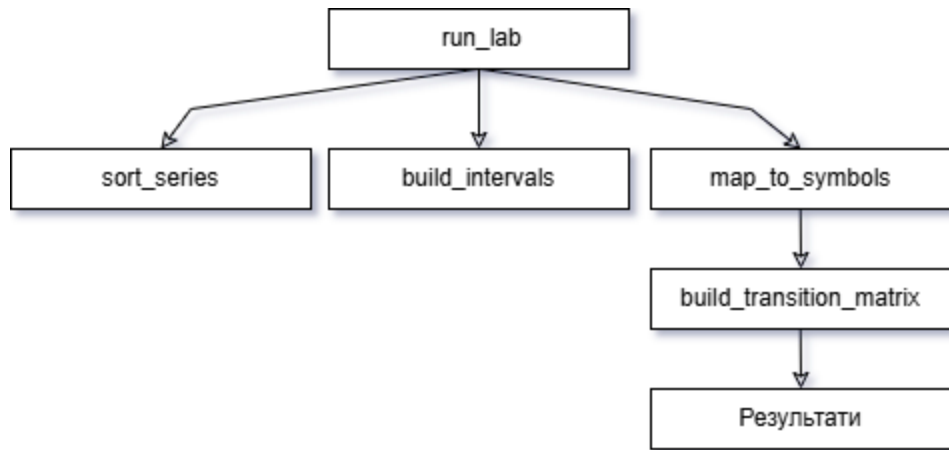
Розподіл ймовірностей: Розбиття на рівні інтервали

Хід розв'язання задачі:

Основні етапи роботи програми:

№	Етап	Опис
1	Задання числового ряду	Ряд чисел задається вручну за допомогою функції <code>c(...)</code> .
2	Сортування ряду	<code>sort_series(series)</code> впорядковує ряд по зростанню для побудови інтервалів.
3	Побудова інтервалів	<code>build_intervals(series, N)</code> розбиває діапазон на рівномірні інтервали згідно з кількістю символів алфавіту.
4	Кодування у символи	<code>map_to_symbols(series, breaks, alphabet)</code> замінює кожне число на символ залежно від інтервалу.
5	Побудова матриці передування	<code>build_transition_matrix(symbols, alphabet)</code> формує таблицю кількості переходів між символами.
6	Вивід результатів	Виводиться лінгвістичний ряд, матриця передування та час виконання.

Функціональна схема



Результати виконання

Перший числовий ряд (5 значень – 3 символи): 3.2, 7.8, 1.5, 9.0, 4.6

Лінгвістичний ряд:

A C A C B

Матриця передування:

A	B	C
---	---	---

A 0 0 2

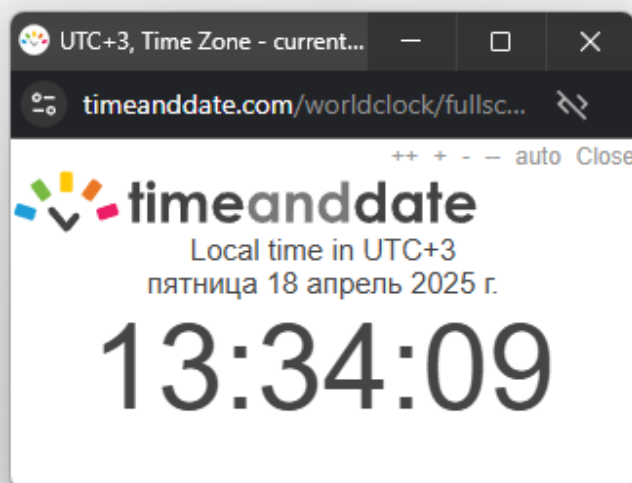
B 0 0 0

C 1 1 0

Час виконання (мілісекунди):

99,217

```
[Execution complete with exit code 0]
```



Другий числовий ряд (5000 значень - 5 символів): B-C-D-E-Gold Futures Historical Data (Price)

Лінгвістичний ряд:

C C C C C C C C C C C C C C C C C C C C C C C D D D D D D D D D D D

Матриця передування:

A B C D E

A 1813 5 0 0 0

B 6 631 14 0 0

C 0 15 937 30 0

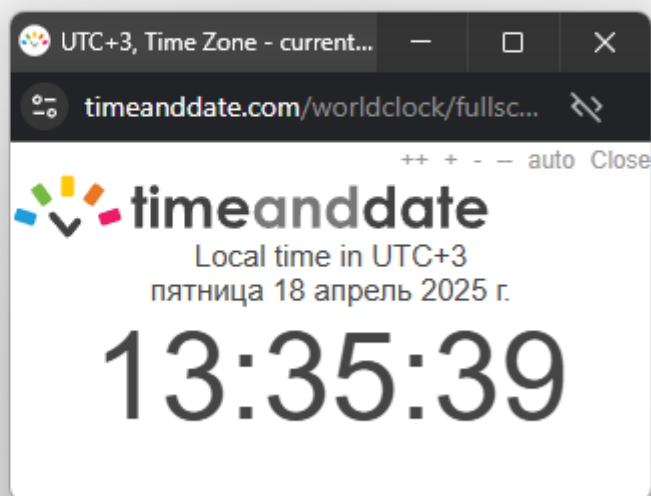
D 0 0 30 1072 9

E 0 0 0 9 428

Час виконання (мілісекунди):

139.016

```
[Execution complete with exit code 0]
```



Третій числовий ряд (5000 значень - 10 символів): B-C-D-E-S&P 500 Historical Data (Price)

Лінгвістичний ряд:

J J J J J J J J I I I J I J J J J J J J J J J J J J J J J J J J J J J J J J J J J J J

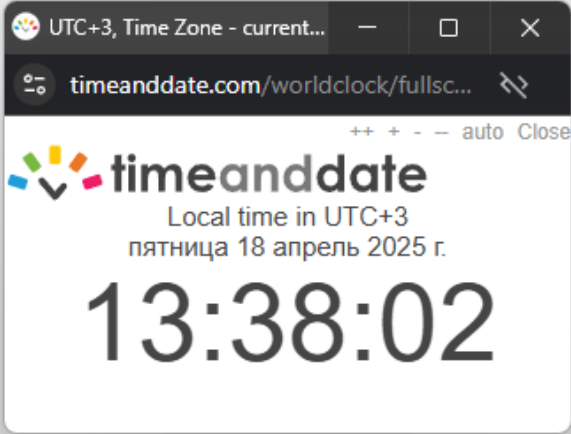
Матриця передування:

[illegible]

Час виконання (мілісекунди):

139.814

```
[Execution complete with exit code 0]
```



Лістинг програмного тексту

```
# === 1. Задаємо числовий ряд ===
series <- c(3.2, 7.8, 1.5, 9.0, 4.6)

# === 2. Визначення функцій ===

# Функція для сортування числового ряду
sort_series <- function(series) {
  sort(series)
}

# Побудова рівномірних інтервалів на основі діапазону чисел
build_intervals <- function(series, alphabet_size) {
  min_val <- min(series)
  max_val <- max(series)
  step <- (max_val - min_val) / alphabet_size
  breaks <- seq(min_val, max_val, by = step)

  if (length(breaks) == alphabet_size) {
    breaks <- c(breaks, max_val)
  }
  return(breaks)
}

# Перетворення чисел у відповідні символи алфавіту
map_to_symbols <- function(series, breaks, alphabet) {
  indices <- findInterval(series, breaks, rightmost.closed = TRUE)

  indices[indices == 0] <- 1
  indices[indices > length(alphabet)] <- length(alphabet)

  return(alphabet[indices])
}

# Побудова матриці передування символів
build_transition_matrix <- function(symbols, alphabet) {
  n <- length(alphabet)
  matrix <- matrix(0, nrow = n, ncol = n,
    dimnames = list(alphabet, alphabet))

  for (i in 1:(length(symbols) - 1)) {
    from <- symbols[i]
    to <- symbols[i + 1]
    matrix[from, to] <- matrix[from, to] + 1
  }
}
```

```

    }
    return(matrix)
}

# Головна функція
run_lab <- function(series, alphabet) {
  alphabet_size <- length(alphabet)
  sorted <- sort_series(series)
  breaks <- build_intervals(sorted, alphabet_size)
  symbols <- map_to_symbols(series, breaks, alphabet)
  transition_matrix <- build_transition_matrix(symbols, alphabet)
  list(symbols = symbols, matrix = transition_matrix)
}

# === 3. Визначення алфавіту ===
alphabet <- c("A", "B", "C")

# === 4. Запуск алгоритму з вимірюванням часу виконання ===
start_time <- Sys.time()
result <- run_lab(series, alphabet)
end_time <- Sys.time()
execution_time <- end_time - start_time

# === 5. Вивід результатів ===
cat("\nЛінгвістичний ряд:\n")
cat(result$symbols, sep = " ")
cat("\n")

cat("\nМатриця передування:\n")
print(result$matrix)

cat("\nЧас виконання (мілісекунди):\n")
cat(round(as.numeric(execution_time, units = "secs") * 1000, 3))

```