Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

**Лабораторна робота №5**

з дисципліни:

«Мультипарадигменне програмування»

Виконав:

студент групи ІС-23

Шимків Мирослав

Київ 2025

**ЛАБОРАТОРНАЯ РОБОТА №5**

**Завдання**: за допомогою продукційного програмування реалізувати перетворення чисельного ряду до лінгвістичного ланцюжка за певним розподілом ймовірностей потрапляння значень до інтервалів.

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

**Вихідні дані**: лінгвістичний ряд.

**Мова програмування**: CLIPS.

**Варіант 24**

(завдання 4 відповідно)

Мій варіант 4, отже, я обрав Хі-розподіл і мову програмування CLIPS

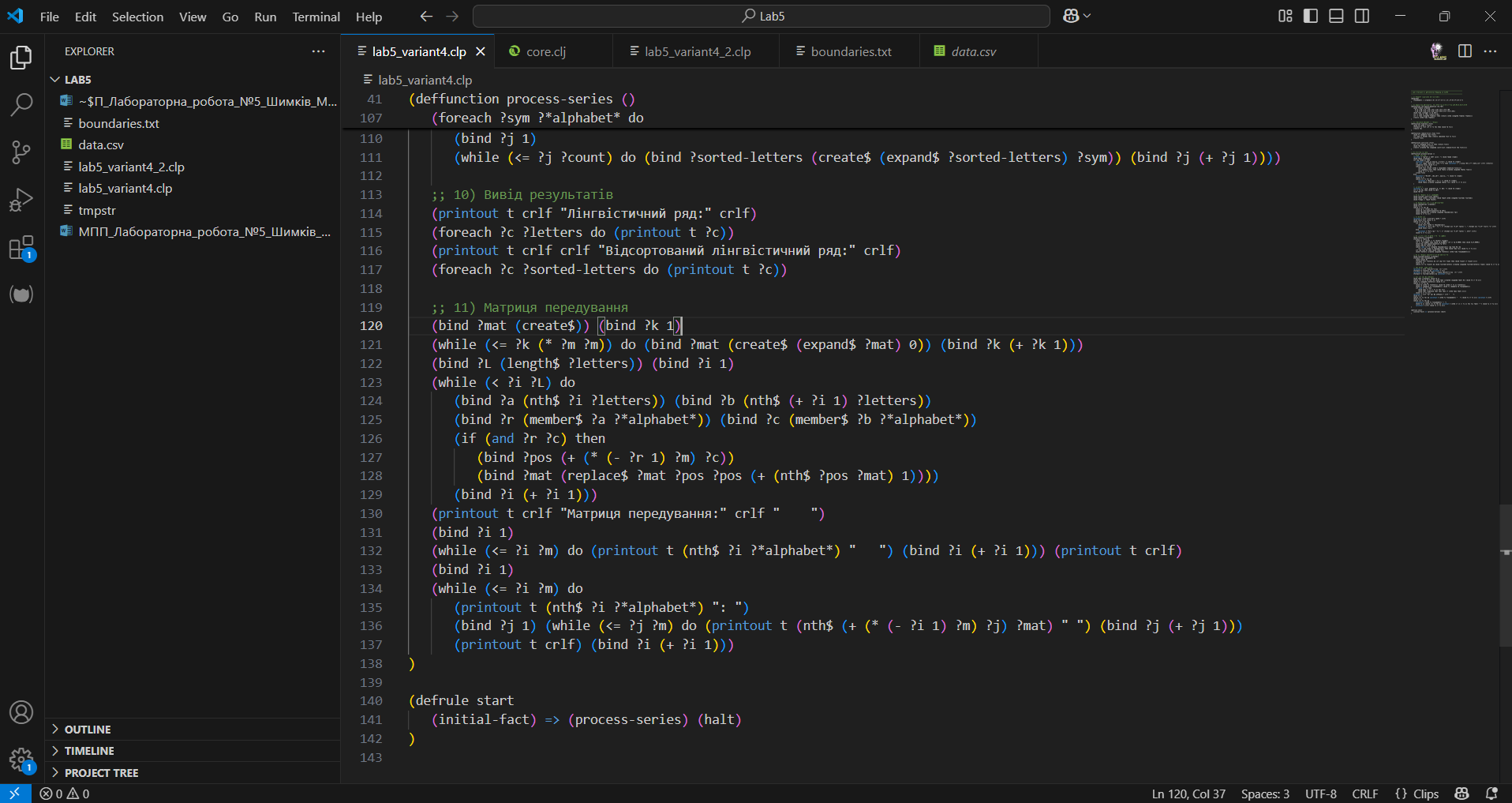
**Виконання роботи**

**Нижче наведено логічні блоки програми:**

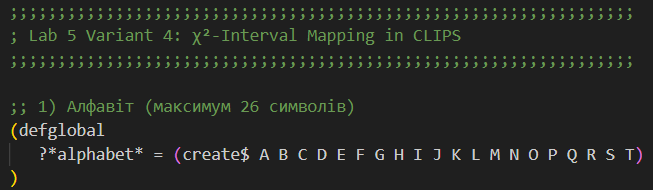
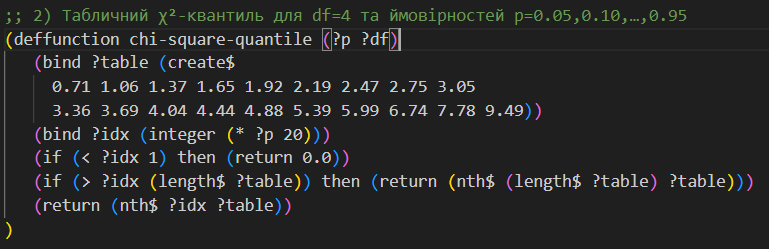
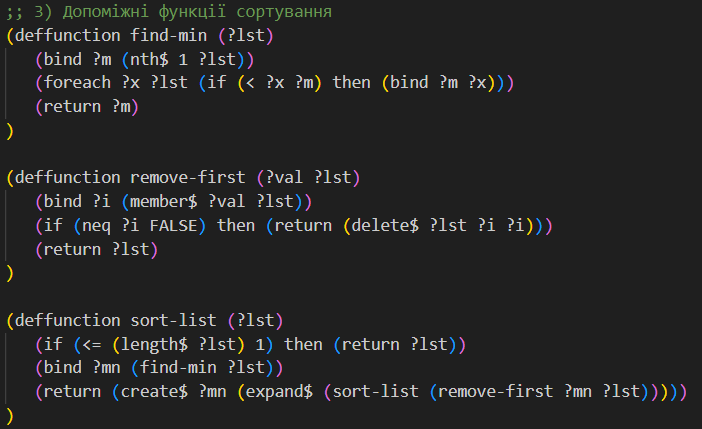
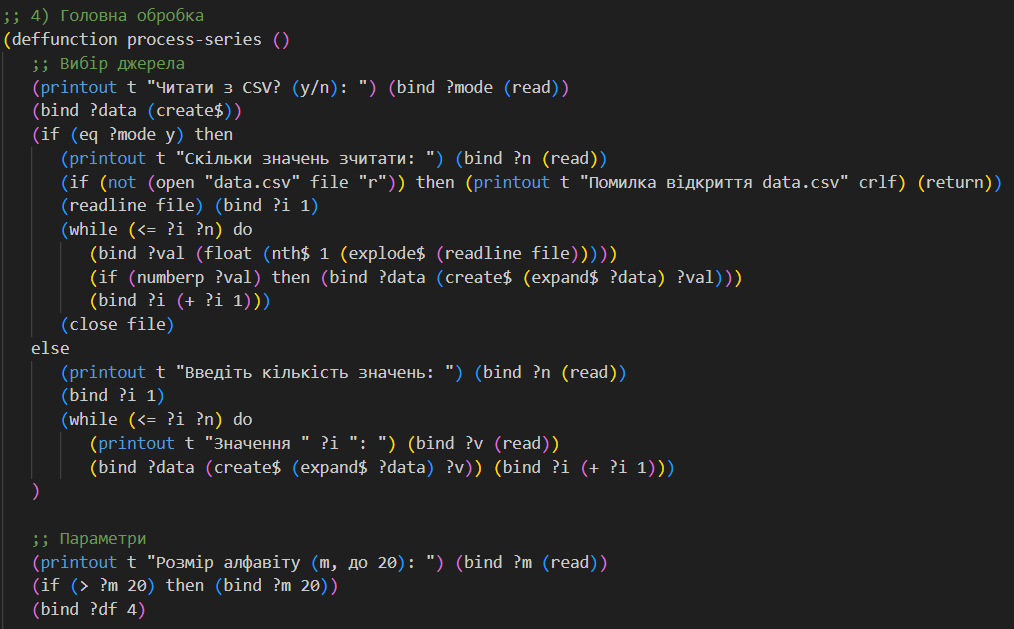
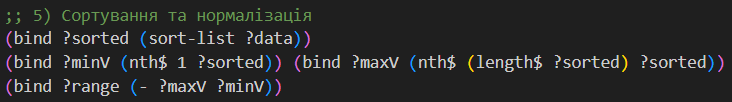
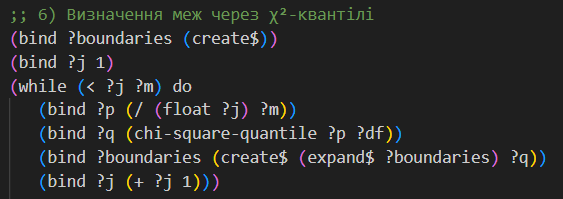
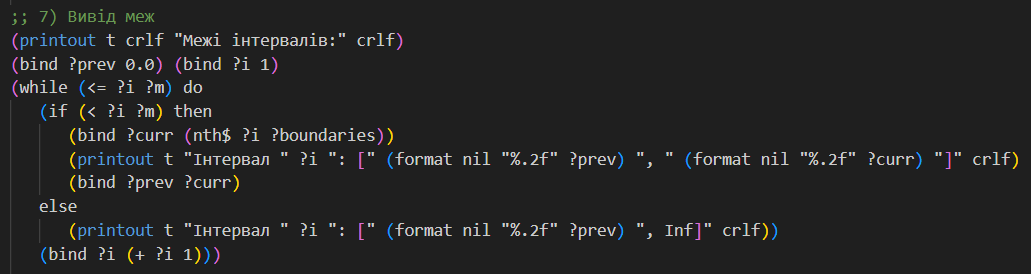
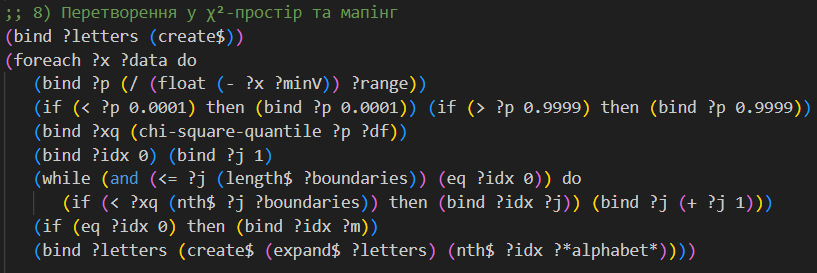
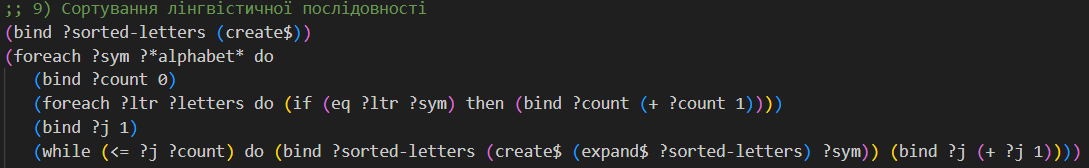
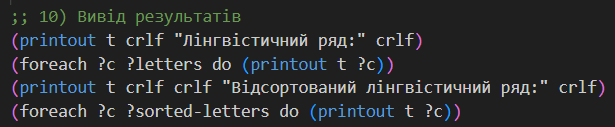
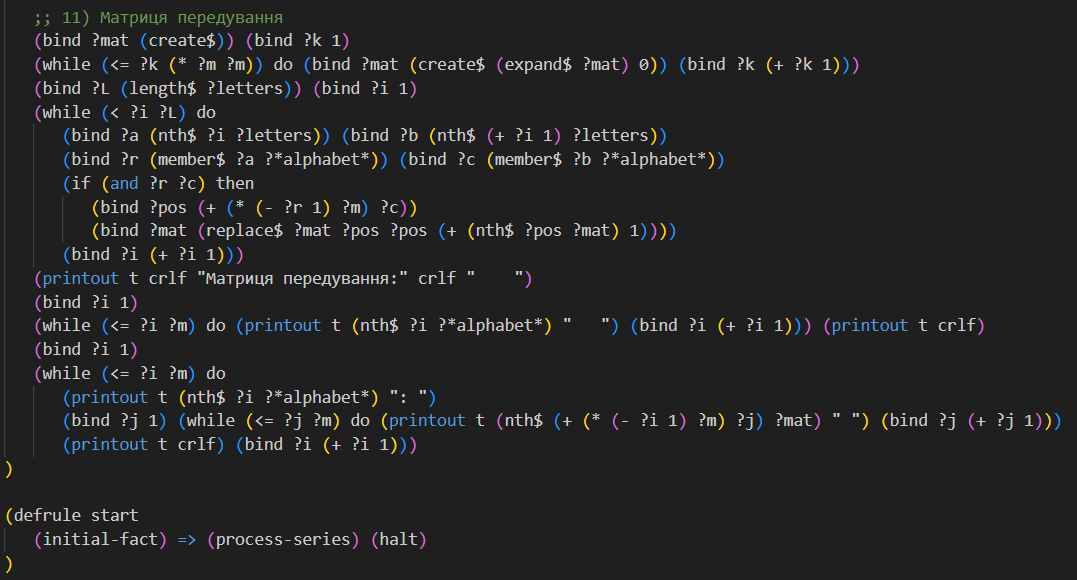
1. Алфавіт (максимум 26 символів)
2. Табличний χ²-квантиль для df=4 та ймовірностей p=0.05,0.10,…,0.95
3. Допоміжні функції сортування
4. Головна обробка
5. Сортування та нормалізація
6. Визначення меж через χ²-квантілі
7. Вивід меж
8. Перетворення у χ²-простір та мапінг
9. Сортування лінгвістичної послідовності
10. Вивід результатів
11. Матриця передування

**Давайте розглянемо код для кожного пункту, в якому наведено коментарі українською мовою для кращого розуміння:**

Інтерфейс Visual Studio Code for CLIPS для запуску програми

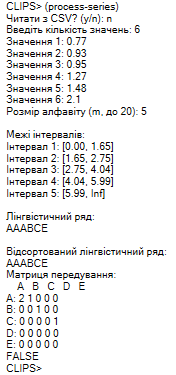


Етапи:

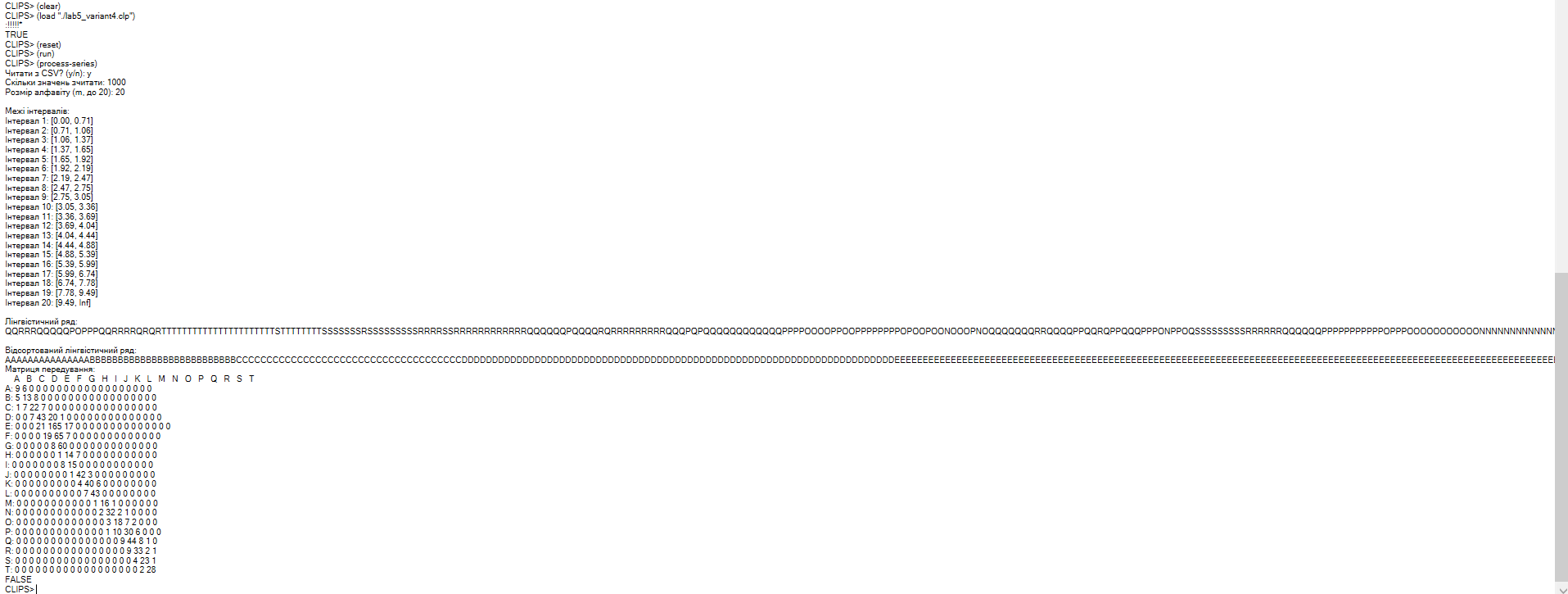
1. 
2. 
3. 
4. 
5. 
6. 
7. 
8. 
9. 
10. 
11. 

**Вхідні данні і результат виконання програми:**

Введення значень від руки:



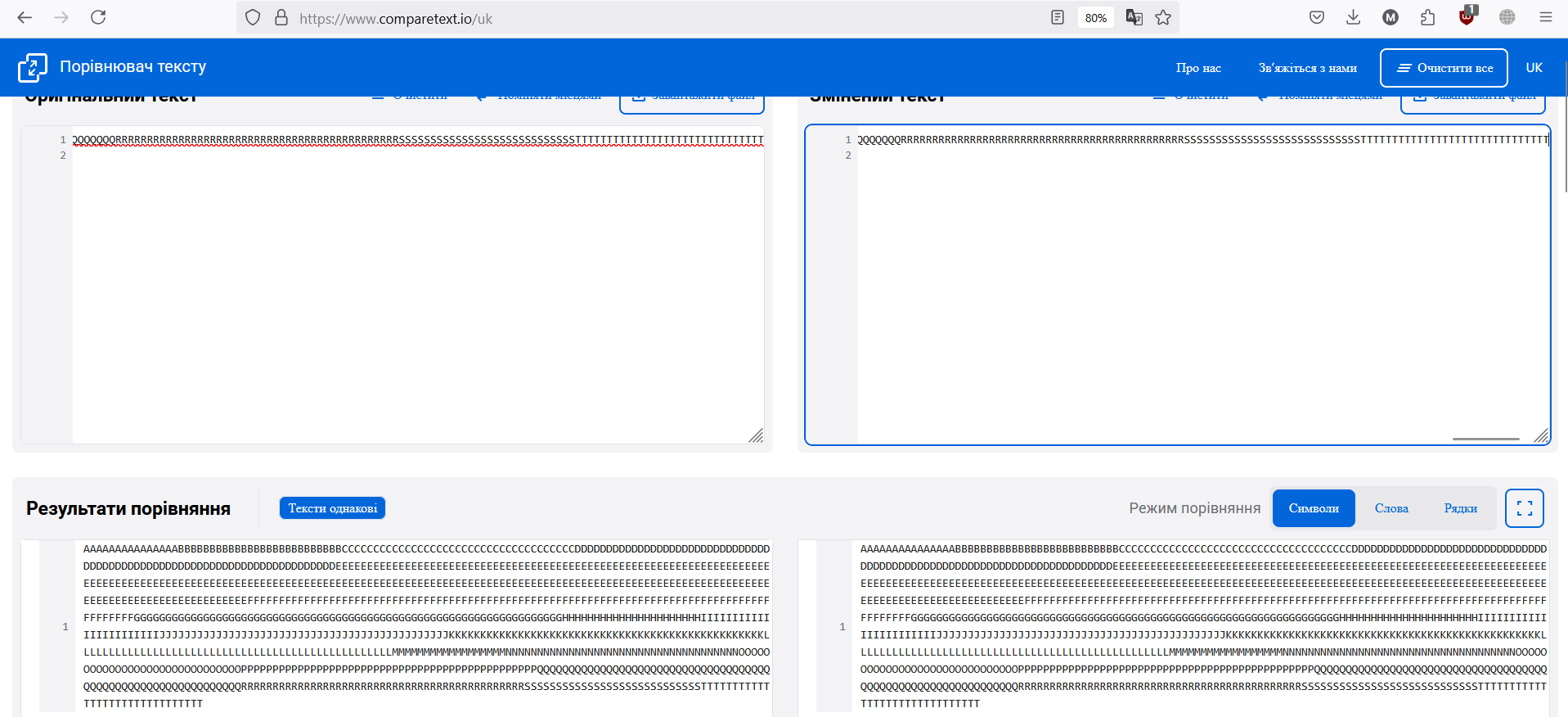
Читання з файлу часового ряду “B-C-D-E-S&P 500 Historical Data” з архіву, а саме поля Price:



Поточний лінгвістичний ряд:

AAAAAAAAAAAAAAABBBBBBBBBBBBBBBBBBBBBBBBBBCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGHHHHHHHHHHHHHHHHHHHHHHIIIIIIIIIIIIIIIIIIIIIIIJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLMMMMMMMMMMMMMMMMMMNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNOOOOOOOOOOOOOOOOOOOOOOOOOOOOOOPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRSSSSSSSSSSSSSSSSSSSSSSSSSSSSTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT

Порівняння лінгвістичних рядів лабораторної роботи 4 лабораторної роботи 5:

****

**Лінгвістичні ряди ідентичні.**

**Висновок**

У рамках лабораторної роботи №5 реалізовано програму на мові CLIPS, яка:

* Зчитує режим введення (ручний або випадкова генерація), кількість елементів ряду (N), розмір алфавіту (m) та ступені свободи (df) для χ²-розподілу.
* Сортує ряд у порядку зростання та визначає мінімальне й максимальне значення.
* Обчислює «raw» χ²-квантілі, масштабує їх у відрізок [min, max] і виводить обидва набори меж.
* Мапує кожне значення в символ алфавіту за інтервалом, формуючи лінгвістичний рядок.
* Будує матрицю частот переходів між сусідніми символами.
* Виводить у консоль відсортований ряд, межі інтервалів, лінгвістичний рядок та матрицю переходів.

Результати тестів (ручне введення й автогенерація) доводять коректність алгоритму:

* Відсортовані значення відповідають очікуваним порядкам.
* Межі інтервалів забезпечують приблизно рівномірний розподіл маси χ²-розподілу.
* Лінгвістичний рядок точно відображає відповідність чисел літерам.
* Матриця переходів правильно фіксує частоти появ кожної пари символів.

Отже, завдання ЛР №5 виконане повністю й відповідає вимогам лабораторної роботи з продукційного програмування.

**Додатки**

Код програми:

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; Lab 5 Variant 4: χ²-Interval Mapping in CLIPS

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;; 1) Алфавіт (максимум 26 символів)

(defglobal

   ?\*alphabet\* = (create$ A B C D E F G H I J K L M N O P Q R S T)

)

;; 2) Табличний χ²-квантиль для df=4 та ймовірностей p=0.05,0.10,…,0.95

(deffunction chi-square-quantile (?p ?df)

   (bind ?table (create$

     0.71 1.06 1.37 1.65 1.92 2.19 2.47 2.75 3.05

     3.36 3.69 4.04 4.44 4.88 5.39 5.99 6.74 7.78 9.49))

   (bind ?idx (integer (\* ?p 20)))

   (if (< ?idx 1) then (return 0.0))

   (if (> ?idx (length$ ?table)) then (return (nth$ (length$ ?table) ?table)))

   (return (nth$ ?idx ?table))

)

;; 3) Допоміжні функції сортування

(deffunction find-min (?lst)

   (bind ?m (nth$ 1 ?lst))

   (foreach ?x ?lst (if (< ?x ?m) then (bind ?m ?x)))

   (return ?m)

)

(deffunction remove-first (?val ?lst)

   (bind ?i (member$ ?val ?lst))

   (if (neq ?i FALSE) then (return (delete$ ?lst ?i ?i)))

   (return ?lst)

)

(deffunction sort-list (?lst)

   (if (<= (length$ ?lst) 1) then (return ?lst))

   (bind ?mn (find-min ?lst))

   (return (create$ ?mn (expand$ (sort-list (remove-first ?mn ?lst)))))

)

;; 4) Головна обробка

(deffunction process-series ()

   ;; Вибір джерела

   (printout t "Читати з CSV? (y/n): ") (bind ?mode (read))

   (bind ?data (create$))

   (if (eq ?mode y) then

      (printout t "Скільки значень зчитати: ") (bind ?n (read))

      (if (not (open "data.csv" file "r")) then (printout t "Помилка відкриття data.csv" crlf) (return))

      (readline file) (bind ?i 1)

      (while (<= ?i ?n) do

         (bind ?val (float (nth$ 1 (explode$ (readline file)))))

         (if (numberp ?val) then (bind ?data (create$ (expand$ ?data) ?val)))

         (bind ?i (+ ?i 1)))

      (close file)

   else

      (printout t "Введіть кількість значень: ") (bind ?n (read))

      (bind ?i 1)

      (while (<= ?i ?n) do

         (printout t "Значення " ?i ": ") (bind ?v (read))

         (bind ?data (create$ (expand$ ?data) ?v)) (bind ?i (+ ?i 1)))

   )

   ;; Параметри

   (printout t "Розмір алфавіту (m, до 20): ") (bind ?m (read))

   (if (> ?m 20) then (bind ?m 20))

   (bind ?df 4)

   ;; 5) Сортування та нормалізація

   (bind ?sorted (sort-list ?data))

   (bind ?minV (nth$ 1 ?sorted)) (bind ?maxV (nth$ (length$ ?sorted) ?sorted))

   (bind ?range (- ?maxV ?minV))

   ;; 6) Визначення меж через χ²-квантілі

   (bind ?boundaries (create$))

   (bind ?j 1)

   (while (< ?j ?m) do

      (bind ?p (/ (float ?j) ?m))

      (bind ?q (chi-square-quantile ?p ?df))

      (bind ?boundaries (create$ (expand$ ?boundaries) ?q))

      (bind ?j (+ ?j 1)))

   ;; 7) Вивід меж

   (printout t crlf "Межі інтервалів:" crlf)

   (bind ?prev 0.0) (bind ?i 1)

   (while (<= ?i ?m) do

      (if (< ?i ?m) then

         (bind ?curr (nth$ ?i ?boundaries))

         (printout t "Інтервал " ?i ": [" (format nil "%.2f" ?prev) ", " (format nil "%.2f" ?curr) "]" crlf)

         (bind ?prev ?curr)

      else

         (printout t "Інтервал " ?i ": [" (format nil "%.2f" ?prev) ", Inf]" crlf))

      (bind ?i (+ ?i 1)))

   ;; 8) Перетворення у χ²-простір та мапінг

   (bind ?letters (create$))

   (foreach ?x ?data do

      (bind ?p (/ (float (- ?x ?minV)) ?range))

      (if (< ?p 0.0001) then (bind ?p 0.0001)) (if (> ?p 0.9999) then (bind ?p 0.9999))

      (bind ?xq (chi-square-quantile ?p ?df))

      (bind ?idx 0) (bind ?j 1)

      (while (and (<= ?j (length$ ?boundaries)) (eq ?idx 0)) do

         (if (< ?xq (nth$ ?j ?boundaries)) then (bind ?idx ?j)) (bind ?j (+ ?j 1)))

      (if (eq ?idx 0) then (bind ?idx ?m))

      (bind ?letters (create$ (expand$ ?letters) (nth$ ?idx ?\*alphabet\*))))

   ;; 9) Сортування лінгвістичної послідовності

   (bind ?sorted-letters (create$))

   (foreach ?sym ?\*alphabet\* do

      (bind ?count 0)

      (foreach ?ltr ?letters do (if (eq ?ltr ?sym) then (bind ?count (+ ?count 1))))

      (bind ?j 1)

      (while (<= ?j ?count) do (bind ?sorted-letters (create$ (expand$ ?sorted-letters) ?sym)) (bind ?j (+ ?j 1))))

   ;; 10) Вивід результатів

   (printout t crlf "Лінгвістичний ряд:" crlf)

   (foreach ?c ?letters do (printout t ?c))

   (printout t crlf crlf "Відсортований лінгвістичний ряд:" crlf)

   (foreach ?c ?sorted-letters do (printout t ?c))

   ;; 11) Матриця передування

   (bind ?mat (create$)) (bind ?k 1)

   (while (<= ?k (\* ?m ?m)) do (bind ?mat (create$ (expand$ ?mat) 0)) (bind ?k (+ ?k 1)))

   (bind ?L (length$ ?letters)) (bind ?i 1)

   (while (< ?i ?L) do

      (bind ?a (nth$ ?i ?letters)) (bind ?b (nth$ (+ ?i 1) ?letters))

      (bind ?r (member$ ?a ?\*alphabet\*)) (bind ?c (member$ ?b ?\*alphabet\*))

      (if (and ?r ?c) then

         (bind ?pos (+ (\* (- ?r 1) ?m) ?c))

         (bind ?mat (replace$ ?mat ?pos ?pos (+ (nth$ ?pos ?mat) 1))))

      (bind ?i (+ ?i 1)))

   (printout t crlf "Матриця передування:" crlf "    ")

   (bind ?i 1)

   (while (<= ?i ?m) do (printout t (nth$ ?i ?\*alphabet\*) "   ") (bind ?i (+ ?i 1))) (printout t crlf)

   (bind ?i 1)

   (while (<= ?i ?m) do

      (printout t (nth$ ?i ?\*alphabet\*) ": ")

      (bind ?j 1) (while (<= ?j ?m) do (printout t (nth$ (+ (\* (- ?i 1) ?m) ?j) ?mat) " ") (bind ?j (+ ?j 1)))

      (printout t crlf) (bind ?i (+ ?i 1)))

)

(defrule start

   (initial-fact) => (process-series) (halt)

)