

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

ЗВІТ  
з лабораторної роботи №6  
з дисципліни «Мультипарадигмне програмування»

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

Київ 2025

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

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

**Вихідні дані:** лінгвістичний ряд.

**Система програмування:** MAXIMA.

**Варіант:** 20

**Розподіл ймовірностей:** Розбиття на рівні інтервали

### **Хід розв’язання задачі:**

Програма спочатку визначає вхідні дані – числовий ряд `data` і набір лінгвістичних символів `alphabet`. Далі за допомогою вбудованої функції `sort` виконується сортування вихідного списку, щоб знайти мінімальне (`min_val`) та максимальне (`max_val`) значення. Різниця між ними ділиться на кількість символів алфавіту `n`, і за отриманою довжиною інтервалу формується список [інтервалів], кожен з яких відповідає одній літері алфавіту.

Після цього кожне число з початкового ряду відображається у лінгвістичний символ за допомогою функції `map_number_to_symbol`: для заданого значення обчислюється індекс інтервалу (заокругленням вниз), і воно асоціюється з відповідною літерою.

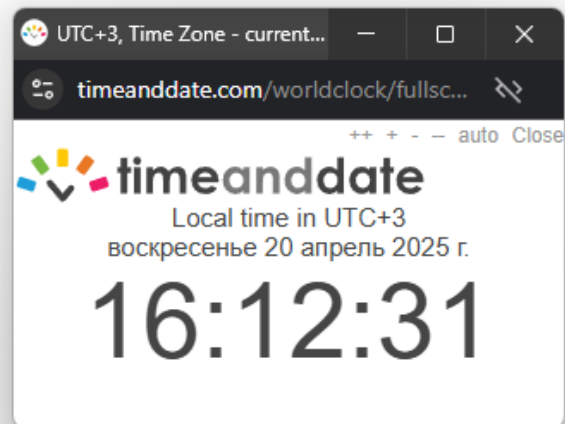
Реалізація виконана в середовищі `Maxima` — системі алгебраїчного програмування з `Lisp`-подібною семантикою. Завдяки високорівневим конструкціям `block` (для локальних змінних та послідовного виконання кроків), `makelist` і `map` (для генерації списків і відображень), а також вбудованим функціям `sort`, `lmin/lmax`, `floor` та роботі зі списками, код залишається компактним і читабельним. Особливістю парадигми є поєднання символічного та числового програмування: можна легко працювати з довільною точністю чисел, динамічно створювати та обробляти списки, а також визначати анонімні функції для гнучкої логіки перетворень.

## Результати виконання

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

```
/* == ВИВІД РЕЗУЛЬТАТІВ == */
print("Лінгвістичний ряд:")$
print(linguistic_sequence)$
```

Лінгвістичний ряд:  
[А, С, А, С, В]



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

```

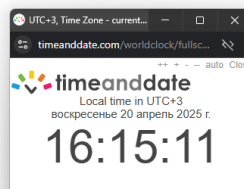
    [idx],
    idx : floor((x - min_val) / interval_length) + 1,
    if idx > n then idx : n,
    alphabet[idx]
  )$
linguistic_sequence : map(map_number_to_symbol, data)$

```

```
/* === ВИВІД РЕЗУЛЬТАТІВ === */
print("Лінгвістичний ряд:")$
print(linguistic_sequence)$
```

Лінгвістичний ряд:

*(The following text is a highly repetitive sequence of letters and symbols, likely representing a corrupted document or a placeholder for a specific pattern.)*



[illegible]

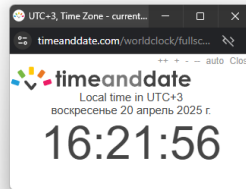
**Третій числовий ряд (5000 значень – 10 символів):** B-C-D-E-S&P 500 Historical Data (Price)

```

        if idx > n then idx := n,
        alphabet[idx]
    )$
    linguistic_sequence := map(map_number_to_symbol, data)$

/* === ВИВІД РЕЗУЛЬТАТІВ === */
print("Лінгвістичний ряд:")$
print(linguistic_sequence)$

```

[illegible][illegible][illegible]

## Лістинг програмного тексту

```
/* === ІНІЦІАЛІЗАЦІЯ ВХІДНИХ ДАНИХ === */
data : [3.2, 7.8, 1.5, 9.0, 4.6]$
alphabet : ["A", "B", "C"]$

/* === СОРТУВАННЯ ЧИСЕЛЬНОГО РЯДУ === */
sorted_data : sort(data)$

/* === РОЗРАХУНОК ІНТЕРВАЛІВ ЗА РОЗМІРНОСТЮ АЛФАВІТУ === */
n : length(alphabet)$
min_val : lmin(data)$
max_val : lmax(data)$
interval_length : (max_val - min_val) / n$
intervals : makelist(
  [ min_val + (i-1)*interval_length,
    min_val + i*interval_length ],
  i, 1, n
)$

/* === ВІДОВРАЖЕННЯ ЧИСЕЛ В ЛІНГВІСТИЧНІ СИМВОЛИ === */
map_number_to_symbol(x) := block(
  [idx],
  idx : floor((x - min_val) / interval_length) + 1,
  if idx > n then idx : n,
  alphabet[idx]
)$
linguistic_sequence : map(map_number_to_symbol, data)$

/* === ВИВІД РЕЗУЛЬТАТІВ === */
print("Лінгвістичний ряд:")$
print(linguistic_sequence)$
```