

## 9. Лабораторна робота. Алгебраїчні обчислення мовою R

**Мета:** засвоїти можливості роботи мови R до застосування розв'язування задач лінійної алгебри.

### Теоретичні відомості та методичні рекомендації

R – є самою популярною у світі мовою для статистичних обчислень. R – це потужна інтерпретована мова сценаріїв для обробки та аналізу статистичних даних, графічної візуалізації даних. Вона широко використовується серед статистиків, дослідників даних та аналітиків завдяки своїй потужності, гнучкості та наявності великої кількості пакетів для вирішення різноманітних завдань.

Ядро R можна завантажити з офіційної сторінки CRAN (Comprehensive R Archive Network) за посиланням для Windows <https://cran.r-project.org/bin/windows/base/>. Крім того, ядро R доступно для інших операційних систем: OS та Linux. Одразу бажано встановити додаткові інструменти для створення пакетів R з вихідного коду в Windows <https://cran.r-project.org/bin/windows/Rtools/> (графічного інтерфейсу немає). Для написання коду можна використовувати програму RStudio IDE (див. рис. 9.1), яка є вільно розповсюдженим IDE-редактором (див. рис. 9.2) коду для мови R <https://posit.co/download/rstudio-desktop/>.

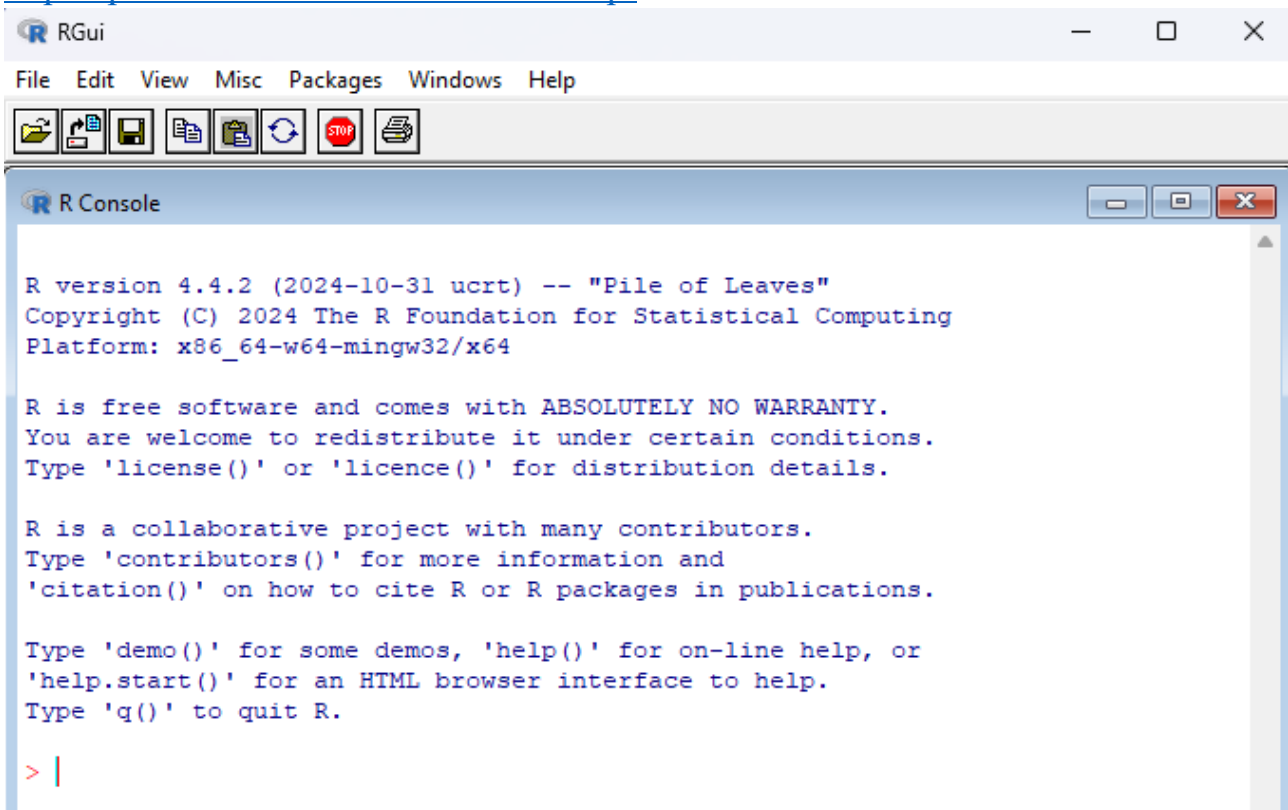


Рисунок 9.1 – Вікно R Console

R Console (рис.) можна використати, наприклад, для оновлення ядра R. Для цього потрібно встановити пакет `installr`, виконавши команди у консолі:

```
install.packages('installr')
library("installr")
updateR()
```

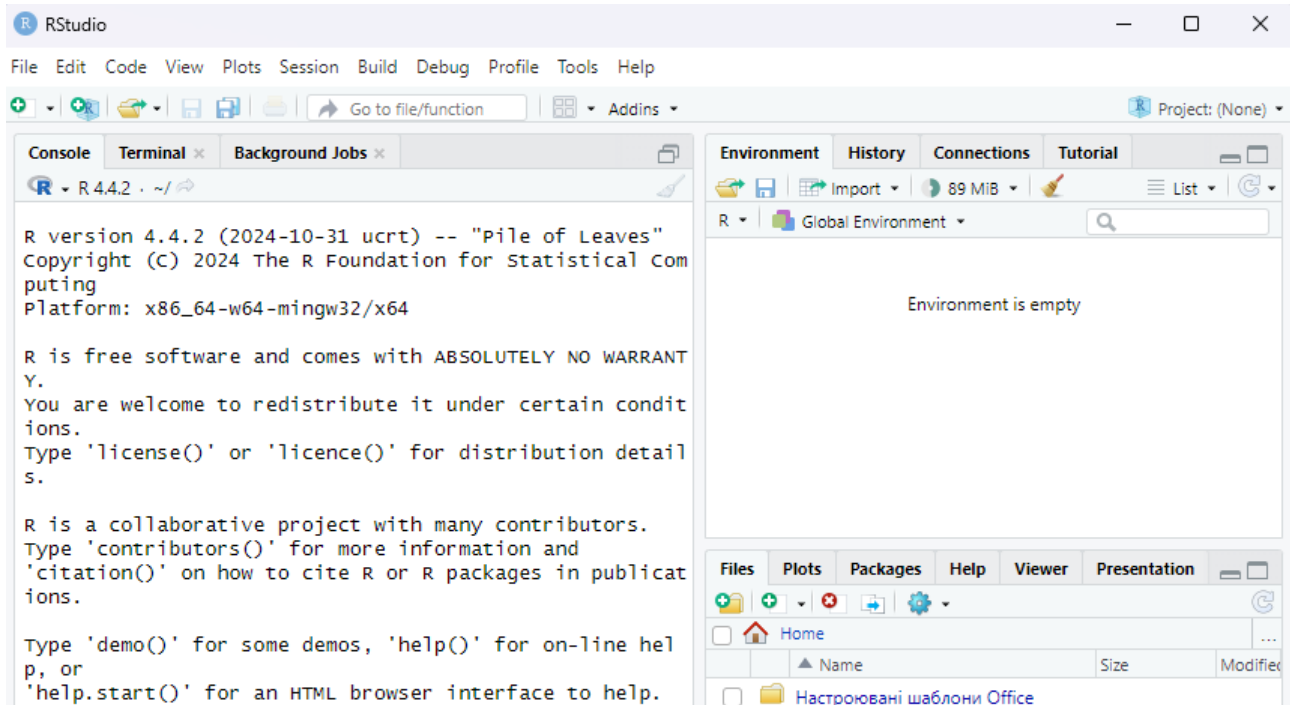


Рисунок 9.2 – Вікно RStudio IDE

Основні характеристики мови R:

- Спрямованість на статистику: R створений спеціально для аналізу даних і включає багато вбудованих функцій для виконання статистичних операцій, таких як регресія, кластеризація, часові ряди тощо.
- Візуалізація даних: R дозволяє створювати якісні графіки, такі як гістограми, діаграми розсіювання, коробкові діаграми, теплові карти тощо. Популярні пакети, як-от `ggplot2`, значно розширюють можливості візуалізації.
- Розширюваність: R має тисячі пакетів, доступних через репозиторій CRAN (Comprehensive R Archive Network). Ці пакети дозволяють працювати з машинним навчанням, біоінформатикою, аналізом тексту, геопросторовими даними та багатьма іншими задачами.
- Відкрите програмне забезпечення: R є безкоштовним і має відкритий вихідний код, що робить його доступним для всіх користувачів і дозволяє спільноті активно розвивати середовище.
- Кросплатформеність: R працює на Windows, MacOS та Linux.
- Скрипти, написані на мові R, зберігаються у форматі `namefile.R`. Команди вводяться користувачем у консолі (командному вікні) після символу запрошення, що має вигляд «>».

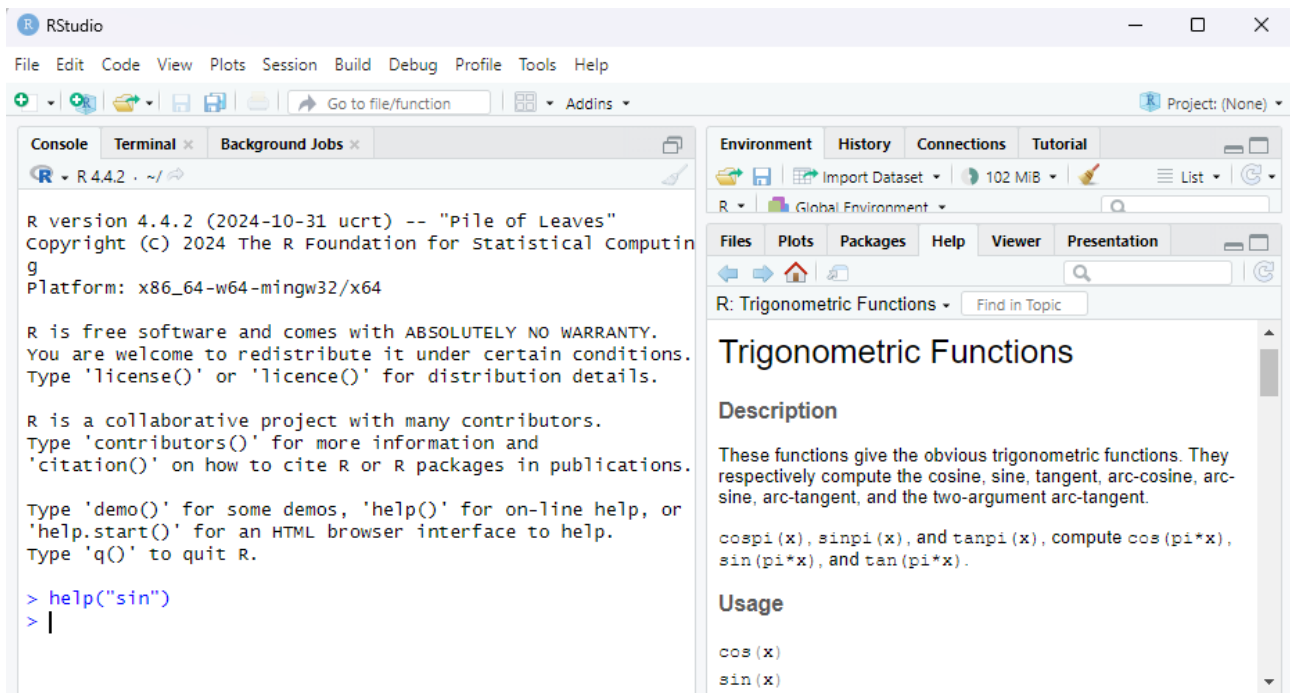


Рисунок 9.3 – Виклик довідки для тригонометричної функції

R чутливий до регістру клавіатури. Всі команди вводяться у командному рядочку. Наприклад, щоб отримати довідку (див. рис. 9.3) за тригонометричною функцією  $\sin x$ , у командному рядочку треба ввести наступне:

```
help("sin") або help.search("sin")
```

У правому нижньому куті на вкладці Help відобразиться інформація щодо виклику функції, аргументу функції та прикладом використання функції.

Основний функціонал програми реалізується за допомогою вбудованих та створених користувачем функцій. Всі об'єкти і набори даних зберігаються у пам'яті до завершення інтерактивної сесії. Основні функції доступні за замовченням. Інші функції містяться у пакетах, які можуть бути активовані у ході роботи з програмою за необхідністю. Командні рядочки складаються з функцій і присвоєнь. R використовує символ  $\leftarrow$  для присвоєння замість звичайного  $=$ . Коментарі пишуться після символу  $\#$ .

Основні поняття:

- **об'єкти**: у R все є об'єктом, включаючи числа, вектори, матриці, списки, дані тощо. Наприклад, матрицю  $2 \times 3$  можна побудувати за допомогою функції `matrix(1:6, nrow=2, ncol=3)`;
- **вектори**: основний будівельний блок даних в R. Вектори можуть містити числа, символи або логічні значення. Вектори в системі R формуються функції конкатенації `c()`;
- **функції**: R має багато вбудованих функцій і дозволяє користувачам створювати свої власні.

Популярні пакети:

- `ggplot2` — для візуалізації даних;

- `dplyr` — для маніпуляції даними;
- `tidyr` — для трансформації даних;
- `caret` — для машинного навчання;
- `shiny` — для створення інтерактивних вебдодатків.

У мові програмування R існують різні типи даних, які дозволяють зберігати і обробляти різноманітну інформацію. Ось основні типи даних у R:

#### **Прості типи даних:**

- `numeric` (числовий). Об'єкти даного класу діляться на цілочислові (`integer`) дійсні (`double`);
- `logical` (логічний). Використовується у логічних умовах. Має два можливих значення: `TRUE` або `FALSE`;
- `complex` (комплексний). Використовується для представлення комплексних чисел. Наприклад: `3 + 2i` (де `i` – уявна одиниця);
- `character` (символьний). Представляє текст або рядки. Рядки записуються у лапках (або в подвійних, або в одинарних). Наприклад: `"hello"`, `'data'`.

#### **Складні типи даних:**

- `vector` (вектор). Одновимірна структура, яка містить елементи одного типу. Приклад: `c(1, 2, 3)` (числовий вектор), `c("apple", "banana")` (символьний вектор);
- `matrix` (матриця). Двовимірна структура, що містить елементи одного типу. Створюється функцією `matrix()`. Наприклад, `matrix(1:6, nrow = 2, ncol = 3)`;
- `array` (масив). Багатовимірна структура, що містить елементи одного типу. Створюється функцією `array()`. Наприклад, `array(1:8, dim = c(2, 2, 2))`;
- `list` (список). Неструктуровані дані, які можуть містити елементи різних типів (числа, рядки, вектори, матриці тощо). Створюється функцією `list()`. Наприклад, `list(1, "text", TRUE, c(1, 2, 3))`;
- `data frame` (фрейм даних). Таблиця, де стовпці можуть містити дані різних типів. Це основний тип для роботи з даними в R. Створюється функцією `data.frame(data1, data2, ...)`. При цьому всі вектори `data1`, `data2`, ... повинні мати однакову довжину. Наприклад,

```
data.frame(
  Name = c("Alice", "Bob"),
  Age = c(25, 30),
  Married = c(TRUE, FALSE)
)
```

- `factor` (фактор). Використовується для представлення категоріальних змінних. Фактор – це векторний об'єкт, що кодує категоріальні дані

(класи). Значення факторів зберігаються як числа, але мають відповідні рівні (labels). Наприклад, `factor(c("low", "medium", "high", "low"))`.

#### Інші типи:

- NULL. Позначає відсутність або неіснування значення. Наприклад, NULL.
- NA (Not Available). Використовується для позначення пропущених значень у даних. Наприклад, NA;
- $Inf$  і  $-Inf$ . Позначають нескінченність. Наприклад,  $\frac{1}{0}$  дасть  $Inf$ ;
- NaN (Not a Number). Позначає результат, який не є числом (наприклад,  $\frac{0}{0}$ ). Наприклад, NaN.

Ці типи (див. рис. 9.4) є основою роботи з даними в R і допомагають ефективно аналізувати та обробляти інформацію.

```
> num <- 3.14
> class(num)
[1] "numeric"
> int <- 42L
> class(int)
[1] "integer"
> log <- TRUE
> class(log)
[1] "logical"
> char <- "Hello, R!"
> class(char)
[1] "character"
> fac <- factor(c("low", "medium", "high"))
> class(fac) # "factor"
[1] "factor"
> levels(fac)
[1] "high" "low" "medium"
>
```

Рисунок 9.4 – Приклад визначення типу даних

Арифметичні операції над матрицями здійснюються по компонентно, тому, наприклад, щоб додати дві матриці, вони повинні мати однакові розміри. Функція `A = cbind(A, B)` створює матрицю, приписуючи до A справа B (для цього кількість рядків у A і B має співпадати). Функція `A = rbind(A, B)` створює матрицю, приписуючи до A знизу B (для цього кількість стовпців у A і B має співпадати).

#### Операції над матрицями.

- $A+B$  – додавання;
- $A-B$  – віднімання;
- $A \%*\% B$  – множення матриць;
- $t(A)$  – транспонування матриці A;
- $\det(A)$  – визначник матриці A;
- `solve(A)` – обернена матриця A;
- `eigen(A)` – знаходження власних значень матриці A.

```

Приклад створення матриці (див. рис. 9.5):
MatrixData_A <- c(1, 2, 3, 4)
A <- matrix(MatrixData_A, nrow=2, ncol=2, byrow =
TRUE)
A

> A
     [,1] [,2]
[1,]    1    3
[2,]    2    4
> |

```

Рисунок 9.5 – Результат виконання скрипту виведення матриці у консолі

### Завдання до лабораторної роботи

Дано матриці  $A$ ,  $B = \frac{1}{v}A$ ,  $C$  згідно варіанту, де  $v$  – номер варіанту.

Напишіть скрипт мовою R, який буде виконувати завдання.

1. Обчислити:  $A + B$ ;  $AB$ ;  $A^{-1}$ ;  $A^T$ ;  $\det(A)$ .
2. Обчисліть власні значення та власні вектори матриці  $A$ .
3. Створіть матрицю  $AB$ , приписуючи до  $A$  справа  $B$  та приписуючи до  $A$  знизу  $B$ .
4. Розв'язати матричне рівняння  $AX = B$  матричним методом.
5. Згенерувати цілочисловий вектор  $M$ , що складається з 50 випадкових чисел, що підкоряються нормальному розподілу з математичним сподіванням 25 і середнім квадратичним відхиленням 10.
6. Визначити максимальний, мінімальний елементи і їхні індекси у векторі  $M$ .
7. Визначити середнє геометричне й медіану значень вектору  $M$ .
8. Визначити моду значень вектору  $M$ .

### Варіанти завдань

$$\begin{aligned}
 1) \quad A &= \begin{pmatrix} -1,7 & 6,8 & 9,4 & 1,3 & 4,6 \\ 2,8 & 0,7 & -5,2 & -9,5 & 12,5 \\ -6,1 & 9,3 & 12,0 & 7,8 & 6,7 \\ 0,5 & -5,2 & -4,1 & -3,4 & -8,4 \\ 1,1 & 11,3 & -5,5 & 2,2 & 7,9 \end{pmatrix}; C = \begin{pmatrix} 25,14 \\ -12,56 \\ 31,22 \\ -17,84 \\ 20,25 \end{pmatrix}. \\
 2) \quad A &= \begin{pmatrix} 3,2 & 3,3 & 4,3 & 1,8 & -8,2 \\ -1,3 & -4,5 & -5,2 & 2,5 & -1,4 \\ 4,9 & 6,3 & 2,0 & 0,8 & 1,9 \\ 0,2 & 8,9 & -14,1 & 6,4 & 0,4 \\ -5,7 & -1,3 & 6,1 & 9,3 & -0,8 \end{pmatrix}; C = \begin{pmatrix} 12,57 \\ -14,01 \\ 26,48 \\ 13,01 \\ 0,85 \end{pmatrix}. \\
 3) \quad A &= \begin{pmatrix} 0,2 & 5,3 & -7,3 & 1,5 & 4,2 \\ 8,5 & -4,5 & 3,8 & -3,8 & -5,5 \\ 6,2 & -0,2 & -13,2 & 0,2 & 6,7 \\ -2,7 & -9,4 & 10,8 & 1,4 & -3,1 \\ -0,9 & 6,5 & 7,3 & -2,3 & -0,1 \end{pmatrix}; C = \begin{pmatrix} 8,32 \\ 5,17 \\ -11,26 \\ 21,09 \\ 28,05 \end{pmatrix}.
 \end{aligned}$$

$$\begin{aligned}
4) A &= \begin{pmatrix} 7,3 & -1,2 & 0,8 & 3,8 & 7,0 \\ 5,9 & -3,1 & -1,8 & -1,9 & 5,2 \\ 5,7 & 11,9 & 17,7 & 6,1 & 6,3 \\ 10,5 & -2,3 & -3,8 & 1,8 & 4,6 \\ 1,6 & 15,1 & -4,9 & 13,1 & 12,7 \end{pmatrix}; C = \begin{pmatrix} -18,91 \\ 12,10 \\ -15,45 \\ 22,18 \\ 17,12 \end{pmatrix}. \\
5) A &= \begin{pmatrix} 2,6 & 13,7 & -3,5 & 11,4 & -4,4 \\ 6,5 & 5,6 & 11,9 & 2,5 & 10,6 \\ 7,7 & 2,1 & -1,2 & 13,2 & 2,1 \\ 3,6 & -1,9 & 7,4 & 1,8 & -7,6 \\ -5,9 & 7,8 & 5,3 & -1,7 & 7,3 \end{pmatrix}; C = \begin{pmatrix} 23,68 \\ 4,18 \\ -15,70 \\ 4,62 \\ -10,98 \end{pmatrix}. \\
6) A &= \begin{pmatrix} -12,7 & 10,1 & 15,4 & -0,5 & -2,6 \\ 11,3 & -3,2 & -7,7 & 12,1 & 5,9 \\ 5,7 & 14,6 & 0,6 & 4,7 & 12,2 \\ 6,4 & 0,8 & -0,1 & 11,6 & 2,7 \\ 13,5 & -0,3 & 14,4 & 11,1 & 9,6 \end{pmatrix}; C = \begin{pmatrix} 14,63 \\ -6,70 \\ -17,16 \\ 7,66 \\ 1,37 \end{pmatrix}. \\
7) A &= \begin{pmatrix} 2,9 & 4,1 & 12,5 & -3,6 & 6,3 \\ 6,5 & 1,4 & 10,3 & -2,3 & -0,5 \\ -4,1 & 0,3 & 7,8 & 6,2 & 5,3 \\ 9,4 & -3,9 & 14,2 & -3,6 & 0,2 \\ -0,4 & 1,6 & 7,5 & 11,2 & 10,6 \end{pmatrix}; C = \begin{pmatrix} -12,92 \\ -2,75 \\ 24,07 \\ 3,58 \\ 23,06 \end{pmatrix}. \\
8) A &= \begin{pmatrix} 11,2 & 7,9 & -4,1 & 2,3 & 13,6 \\ 14,1 & 9,6 & 4,1 & 11,7 & -9,3 \\ -12,4 & -0,2 & 10,3 & 14,9 & 8,8 \\ 2,5 & 1,1 & 6,6 & 5,9 & 3,4 \\ -1,9 & 1,8 & 9,5 & 4,7 & 8,2 \end{pmatrix}; C = \begin{pmatrix} 10,46 \\ 12,11 \\ -9,42 \\ -16,01 \\ 10,24 \end{pmatrix}. \\
9) A &= \begin{pmatrix} 8,8 & 14,6 & -2,2 & 9,1 & 4,7 \\ 4,8 & 0,7 & -3,7 & 10,2 & -0,3 \\ -4,2 & 12,9 & 10,1 & 7,8 & 8,9 \\ 5,9 & 5,1 & 14,7 & 2,6 & 13,6 \\ 1,7 & 9,5 & 4,7 & 4,2 & -2,6 \end{pmatrix}; C = \begin{pmatrix} 18,04 \\ 5,78 \\ -7,45 \\ -15,36 \\ 18,25 \end{pmatrix}. \\
10) A &= \begin{pmatrix} -2,2 & 6,4 & 13,4 & 8,5 & 3,9 \\ 3,7 & 2,1 & -5,5 & 7,2 & 7,1 \\ 12,7 & 8,2 & -9,1 & 8,9 & 3,4 \\ 2,1 & 15,6 & -3,6 & 11,4 & 12,2 \\ -14,1 & 7,9 & -5,3 & 15,2 & 11,7 \end{pmatrix}; C = \begin{pmatrix} 4,98 \\ -13,61 \\ -17,36 \\ 6,59 \\ -9,82 \end{pmatrix}.
\end{aligned}$$

### Питання для самоконтролю

1. Для чого використовують мову R?
2. Для чого потрібен Rtools?
3. Які Ви знаєте популярні пакети в R. Охарактеризуйте їх.
4. Охарактеризуйте команди введення\виведення в R.
5. Охарактеризуйте типи даних в R.
6. Наведіть приклади основних команд для роботи з типами даних.
7. Охарактеризуйте структури даних в R.

8. Наведіть приклади використання основних команд для роботи зі структурами даних.
9. Опишіть способи організації функцій в R.
10. Наведіть приклади Data Frame.
11. Як задаються матриці на мові R?
12. Охарактеризуйте операції над матрицями.