

*Чіома Е. В., студентка 4-го курсу
спеціальності 122 «Комп'ютерні науки»
Нескородева Т. В., к.т.н, доцент, завідувач
кафедри інформаційних технологій*

МОВА R: ОСОБЛИВОСТІ ВИКОРИСТАННЯ ЦИКЛІВ

Донецький національний університет імені Василя Стуса, м. Вінниця

R – це мова програмування і програмне середовище для статистичних обчислень, аналізу та зображення даних в графічному вигляді [1]. Розробка R відбувалась під впливом двох наявних мов програмування: мови програмування S з семантикою успадкованою від Scheme. R названа за першою літерою імен її засновників Роса Іхаки (Ross Ihaka) та Роберта Джентлмена (Robert Gentleman) працівників Оклендського Університету в Новій Зеландії.

R має значні можливості для здійснення статистичних аналізів, включаючи лінійну і нелінійну регресію, класичні статистичні тести, аналіз часових рядів (серій), кластерний аналіз тощо. R легко розбудовується завдяки використанню додаткових функцій і пакетів доступних у Comprehensive R Archive Network (CRAN). Більша частина стандартних функцій R, написана мовою R, однак існує можливість підключати код написаний C, C++, або Фортраном [2].

В мовах для роботи з великим обсягом даних, по типу R – цикли, як правило, поступаються ефективності спеціалізованим функціям запитів, фільтрації, агрегації та трансформації даних.

У багатьох мовах програмування цикли служать базовими блоками, які використовуються для будь-яких завдань, що повторюються. Однак надмірне або неправильне використання циклів в R може призвести до відчутного падіння продуктивності — при тому, що способів написання циклів у цій мові надзвичайно багато.

R підтримує такі основні класичні способи написання циклів:

- **for** – найпоширеніший тип циклів. Синтаксис дуже простий і знайомий розробникам різних мов програмування. For виконує передану йому функцію кожного елемента.

```
# Виведемо номери від 1 до 10
for(i in 1:10)
  print(i)

# Виведемо всі рядки із вектора strings
strings <- c("Один", "Два", "Три")
for(str in strings)
  print(str)
```

Трохи менш поширені `while` та `repeat`, які теж часто зустрічаються в інших мовах програмування.

- У `while` перед кожною ітерацією перевіряється логічна умова, і якщо вона дотримується, то виконується ітерація циклу, якщо ні цикл завершується:

```
while(cond) expr
```

- У `repeat` цикл повторюється доти, доки у явному вигляді не буде викликаний оператор `break`:

```
repeat expr
```

Варто відзначити, що `for`, `while` і `repeat` завжди повертають `NULL`, в цьому їхня відмінність від інших групи циклів.

- Цикли на основі `apply` (`apply`, `earply`, `lapply`, `mapply`, `rapply`, `sapply`, `tapply`, `vapply`) – це досить великий перелік функцій циклів, об'єднаних однією ідеєю. Відрізняються вони тим, що при виклику даних функцій зазначаються параметри до яких цикл застосовується і який результат повинен повертати. Базовий цикл `apply`, який застосовується до матриць:

```
apply(X, MARGIN, FUN, ...)
```

У першому параметрі (`X`) вказується вихідна матрицю, у другому параметрі (`MARGIN`) уточнюється спосіб обходу матриці (1 - по рядках, 2 - по стовпцях, 3 (1,2) - по рядках і стовпцях), третім параметром вказується функція `FUN`, яка буде викликана для кожного елемента. Результати всіх викликів будуть об'єднані в один вектор або матрицю, яку функція `apply` поверне як результат [3].

Цикли, засновані на `apply`, відрізняються від класичних тим, що вони повертають результат роботи циклу, що складається з результатів кожної ітерації, що суттєво прискорює виконання запиту.

- `foreach` – не базова для мови R функція. Відповідний пакет необхідно встановити та перед викликом підключити:

```
install.packages("foreach") # Установка пакету на комп'ютер (один раз)
library(foreach)           # Підключення пакету
```

Незважаючи на те, що `foreach` – стороння функція, на сьогоднішній день це є дуже популярний підхід до написання циклів. `Foreach` був розроблений однією з найвідоміших у світі R компанією – `Revolution Analytics`, яка створила свій комерційний дистрибутив R. У 2015 році компанія була куплена `Microsoft`, і зараз її напруцювання входять до складу `Microsoft SQL Server R Services`. Втім, `foreach` є звичайним `open source` проектом під ліцензією `Apache License 2.0`. Основні причини популярності `foreach`:

- `foreach` повертає значення, які збираються з результатів кожної ітерації, при цьому можна визначити свою функцію та реалізувати будь-яку логіку збору фінального значення циклу з результатів ітерацій;
- є можливість використовувати багатопоточність та запускати ітерації паралельно.

Однією з головних переваг `foreach` є легкість переходу від послідовної обробки до паралельної.

Працюючи з великим обсягом даних цикли не завжди виявляються кращим вибором. Використання спеціалізованих функцій для вибірки, агрегації та трансформації даних завжди ефективніше за цикли. R пропонує безліч варіантів реалізації циклів. Основна відмінність класичних `for`, `while` та `repeat` від групи функцій на основі `apply` полягає в тому, що останні повертають значення [4].

Використання циклів `foreach` з однойменного зовнішнього пакета дозволяє спростити написання циклів, гнучко оперувати значеннями, що повертаються ітераціями, а за рахунок підключення багатопоточної обробки ще й суттєво збільшити продуктивність рішення.

Список літературних джерел

1. Нескородева Тетяна Василівна Method for Automatic Analysis of Compliance of Expenses Data and the Enterprise Income by Neural Network Model of Forecast / Нескородева Тетяна Василівна. // CEUR Workshop Proceedings . – 2020. С.
2. Учасники проєктів Вікімедіа. R (мова програмування) – Вікіпедія. Вікіпедія. URL: [https://uk.wikipedia.org/wiki/R_\(мова_програмування\)](https://uk.wikipedia.org/wiki/R_(мова_програмування)) (дата звернення: 05.12.2021).
1. Машинное обучение. Разработка на R: тонкости при использовании циклов. «Хакер» – Безопасность, разработка, DevOps. URL: <https://haker.ru/2017/01/20/loops-in-r-v2/> (дата звернення: 05.12.2021).
3. Schvepsss. Разработка на R: тайны циклов. Хабр. URL: <https://habr.com/ru/company/microsoft/blog/320232/> (дата звернення: 05.12.2021).