

3) Lévinas E. Totalité et infini. Essai sur l'extériorité / Emmanuel Lévinas. – Paris, LGF, Le Livre de poche, coll. «Biblio-essais», 1990.

4) Merleau-Ponty M. Phénoménologie de la perception / M. Merleau-Ponty. – Paris: Gallimard, 1945. – 531 p.

## **Специфіка програмного коду як тексту**

*Олександр Манделіна*

Київський національний університет імені Тараса Шевченка  
(Київ)

Програмний код є структурним елементом нових медіа, невід'ємною частиною сучасної технологічної реальності, що існує у невидимий для користувача спосіб. Оскільки основою цієї нової реальності є цифрові технології, необхідним виявляється звернення до таких предметів дослідження, що є більш звичними для комп'ютерних наук. Нові медіа визначаються як такі, що є програмовними, це одна з їхніх ключових характеристик. Більше того, «програмування є первинним способом взаємодії з комп'ютером як із річчю» [Куртов 2014, с. 7]. Програмовність є невідчужуваною для нових медіа, оскільки ключовий медіум гіперсфери – це комп'ютер, який принципово визначається як єдність апаратної та програмуючої частини. Така програмовність здійснюється за допомогою мов програмування, які слугують посередником і засобом комунікації як між розробником і комп'ютером, так і між самими розробниками. На цих мовах пишуться тексти у вигляді програмного коду, який, в силу своєї динамічності, постає як мовлення. Технічні системи, що нас оточують, працюють за допомогою програмного забезпечення, а їхня поведінка прописана програмним кодом. І «хоча ці системи говорять різними мовами і переслідують різні цілі, вони поділяють спільний програмний синтаксис/код» у вигляді таких спільних моментів, як оператори управління, структури і типи даних, умовні оператори тощо [Манович 2011]. Тобто є сенс говорити про програмний код як окреме явище і досить визначену область потенційного аналізу. Тому, якщо ми прагнемо зрозуміти логіку нових медіа, нам необхідно звернутися до програмного

коду як мови програмного забезпечення і значною мірою нових медіа в цілому. Логіка, про яку йдеться, визначає специфіку об'єктів та процесів цифрової культури, містить в собі ключі для пояснення її розвитку.

Програмний код породжує специфічний вид текстуальності, що якісно відрізняється від попередніх. Однією з визначальних таких відмінностей є перформативність. Якщо текст такої медіатехнології як книга був статичним і здійснювався у вигляді двох операцій – читання та письма, то для коду додається третя – виконання [Киттлер 2015]. Він є не лише інструкцією, вказівкою до дій, але й тим алгоритмом, що безпосередньо виконується машиною і не може розглядатися поза цією властивістю: можливість бути виконаним є критерієм його коректності. Крім того, перформативність має місце й на іншому рівні: це текст, що перетворюється на операцію над матеріальною дійсністю, що є певною мірою реалізацією давніх магічних сподівань.

Код має процесуальний характер, діє як алгоритм, покликаний відповідати на питання «як?», а не «що?». Як зауважив Ф. Китлер, «знаки перетворюються в операції над знаками»: програмний код як знакова система покликаний оперувати іншими знаками, на нижчому рівні, аж до бінарного коду, який вже є зрощеним із апаратною частиною [Киттлер 2015, с. 183].

Код програми не вичерпується самим текстом як послідовністю знаків – він має багаторівневу архітектуру, що містить в собі ряд моделей та концепцій, які не прочитуються безпосередньо. Спосіб написання та застосування коду визначається абстракціями, патернами, що є невід'ємними від нього і знаходяться у такому ж зв'язку, як, власне, будівельна архітектура та структура думки епохи, що її породила. Так, «предметом думки програміста є невидиме як таке» [Куртов 2014, с. 29]. Програмний продукт не піддається вичерпній візуалізації, він принципово «невидимий та невізуалізований», оскільки «реальність програмного забезпечення не вбудовується природнім чином у простір», – стверджує Фредерік Брукс [Брукс 1999, с. 169]. Неможливо візуалізувати структуру програми, що «прописується» кодом, та всі прошарки програмної абстракції у їх складній взаємодії.

Програмному коду властива також «самоописовість», яка полягає у тому, що він містить «висловлювання, сформульовані не

лише об'єктною мовою, але й метамовою», тобто такі блоки тексту, які не призначені для виконання, а слугують описом і поясненням, що може мати принципове значення для правильної інтерпретації [Манович 2015]. Більше того, самоописовість коду має місце й на значно глибшому рівні, оскільки він вже сам по собі є описом способу дії, яка у своїй абстрактній есенції є, фактично, абстрактною машиною. У цьому сенсі він є сутністю, яка сама є своїм описом. Зразком цього явища можна вважати універсальну дискретну машину (машина Тюрінга), яка є елементарним комп'ютером і лежить в основі всіх електронно-обчислювальних машин. Вона настільки збігається з її описом, що більш повно і точно описати її неможливо [Киттлер 2015]. Звичайно, ми не можемо звести розуміння програмного коду до подібного самоопису, але подібне явище має місце, оскільки текст програми представляє собою опис її роботи і становить її «тіло», що має сенс вже тільки в межах самого коду.

#### Література:

- 1) Брукс Ф. Мифический человеко-месяц, или Как создаются программные системы / Фредерик Брукс. – СПб: Символ-Плюс, 1999.
- 2) Киттлер Ф. Медиа философии, философия медиа / Фридрих Киттлер // Логос. – 2015. – №2 (104). – С. 173–188.
- 3) Куртов М. Генезис графического пользовательского интерфейса. К теологии кода [Электронный ресурс] / Михаил Куртов // ТрансЛит. – 2014. – Режим доступа: <https://www.academia.edu/7410230/>.
- 4) Манович Л. Как следовать за пользователями программ? / Лев Манович // Логос. – 2015. – №2 (104). – С. 189–218.
- 5) Манович Л. Культурный софт [Электронный ресурс] / Лев Манович / пер. Е. Арье, О. Мороз. – 2011. – Режим доступа: [http://cultlook.org/lm\\_cultural\\_software](http://cultlook.org/lm_cultural_software).