

Wprowadzenie do studiów nad oprogramowaniem

Mirostław Filiciak

ZAJRZEĆ POD POWIERZCHNIĘ EKРАНU

O programowanie modeluje rzeczywistość – to teza, z którą stosunkowo łatwo jest się zgodzić. Algorytm wyszukiwarki internetowej i rekomendacje z witryny sieciowej księgarni coraz częściej określają to, co czytamy, a nasze wybory równocześnie służą algorytmowi Google do tworzenia własnych rankingów (Google jest zresztą w stanie na podstawie wpisywanych w wyszukiwarkę hasel przewidzieć np. rozwój epidemii grypy co najmniej o tydzień wcześniej, niż odpowiedzialne za to państwowe służby medyczne). Funkcje zawarte w edytorach tekstu wpływają na kształt tego, co piszemy. Oprogramowanie wykorzystywane już chyba we wszystkich sferach ludzkiej kreatywności wpływa na to, jaki kształt przyjmują obrazy, na które patrzymy, i dźwięki, których słuchamy. Wielki biznes i wojsko podejmują decyzje korzystając ze wsparcia komputerów. Oprogramowanie coraz częściej podsuwa też metafory, którymi „żywi się” kultura – począwszy od nowych modeli autorstwa i produkcji kulturowej adoptowanych z ruchu Open Source, na środkach opisu kończąc („kultura odczytu i zapisu”, „kultura copy/paste”). Oprogramowanie stanowi integralną, choć często niewidoczną, składową naszej codzienności i jest być może tym, co w największym stopniu odpowiada za kształt współczesnej ekonomii, zorientowanej na globalny przepływ danych, usługi i manipulację symbolami. Za sprawą oprogramowania nie tylko pojawiły się nowe obiekty kultury, ale zmieniła się też logika funkcjonowania ich poprzedników. Języki programowania w pewnym sensie wyznaczają granice naszego świata.

Współczesna humanistyka dostarcza wielu dowodów na fundamentalny (co nie zawsze oznacza, że rewolucyjny) charakter zmian, spowodowanych rozwojem technologii informatycznych. Podstawowym konceptem teoretycznym, w którym znalazły one wyraz, jest społeczeństwo sieci. Społeczeństwo sieci organizuje się wokół informacji – jak pisze Manuel Castells „komputery, systemy komunikacyjne i genetyczne dekodowanie oraz programowanie stają się (...) wzmocnieniami i ekstensjami ludzkiego umysłu” (Castells, 2007: 46). Sieć stanowi, według Castellsa, podstawowy model organizacji społecznej, wyznacza nową logikę funkcjonowania jednostek. Wpływ sieci jako formy organizacyjnej, ale i zjawiska wzmacniającego rolę komputerów i komunikacji elektronicznej w naszym życiu jest trudny do przecenienia i rozciąga się nawet na sfery, takie jak prawo (jak napisał Lawrence Lessig, „rządzi raczej kod, nie prawo”) (2005: 175). Zmieniły się praktyki odbioru mediów, dotychczasową widownię zastąpiły bardziej aktywne – przynajmniej w zasobnej w kapitał kulturowy części społeczeństwa – usieciowione jednostki, rzucające wyzwanie tradycyjnemu podziałowi na producentów i konsumentów (zob. Varnelis 2008). Przyniesione spostrzeżenia nie przełożyły się jednak na radykalną zmianę perspektywy teoretycznej, stworzenie innego niż dotychczas podejścia do badania społeczeństwa i kultury. Czy powinny? Według przedstawicieli nowego, wywodzącego się z medioznawstwa nurtu „studiów nad oprogramowaniem” – tak.

Czym jest lub czym ma być *software studies*? Na chwilę obecną nie sposób udzielić na to pytanie jednoznacznej odpowiedzi, bo jak większość nowych dyscyplin i ta chętniej wytyka innym popełnione błędy, niż odpowiada wprost, jak je naprawić. Fundament studiów nad oprogramowaniem stanowi założenie, że tradycyjne podejście do badania mediów, obok właściwości

środków przekazu skoncentrowane na przekazywanych w nich tekstach i tym, co robią z tymi tekstami odbiorcy, nie wystarcza – jest opisem efektów, a nie przyczyn. Pomija bowiem kod komputerowy, ukrytą warstwę warunkującą funkcjonowanie środków przekazu. To właśnie oprogramowanie ma stanowić klucz do zrozumienia świata mediów cyfrowych – jak napisał Lev Manovich: „Żeby zrozumieć nowe media, musimy odwołać się do informatyki. To właśnie tam znajdziemy nowe terminy, kategorie i funkcje charakteryzujące media, które stały się programowalne. Od medioznawstwa zmierzamy w stronę czegoś, co można by nazwać programoznawstwem, czyli od teorii mediów do teorii oprogramowania” (Manovich, 2006: 117). Ten fragment *Języka nowych mediów* położył w pewnym sensie podstawy pod nową dyscyplinę. Manovich – zapewne odpowiadając na zarzuty pomijania roli czynnika ludzkiego, w swej najnowszej, udostępnionej niedawno w sieci książce o znaczącym tytule *Software Takes Command*, uzupełnia: „Software studies muszą zająć się zarówno rolą oprogramowania w kształtowaniu współczesnej kultury, jak i czynnikami kulturowymi, społecznymi oraz ekonomicznymi, które wywierają wpływ na rozwój oprogramowania” (Manovich, 2008c: 6). Mamy więc do czynienia z próbą utworzenia nowego paradygmatu badawczego, interdyscypliny, w której znajdzie się miejsce i dla starszych nauk humanistycznych (o ile zainteresują się oprogramowaniem), która jednak stawia sobie za cel stworzenie własnej metodologii.

Swoje programoznawstwo Manovich obecnie próbuje osadzić w polu akademickim także na poziomie instytucjonalnym, uruchamiając program Software Studies Initiative na Uniwersytecie Kalifornijskim w San Diego. Wobec takiego umocowania studia nad oprogramowaniem wchłoną zapewne inne propozycje związane z analizą oprogramowania komputerowego, jak np. ideę krytycznych studiów nad kodem (*critical code studies*) Marka C. Marino, proponującego czytanie i interpretację kodu komputerowego w sposób do tej pory praktykowany w odniesieniu do literatury (zob. Marino, 2006), czy postulat stworzenia krytyki oprogramowania (*software criticism*), którego próbkę zaprezentował Matthew Fuller w swojej książce *Behind the Blip*, analizując między innymi działanie wyszukiwarek internetowych czy edytora tekstów Word z perspektywy krytycznego użytkownika, którego wzrok nie zatrzymuje się na powierzchni ekranu, lecz wnika głębiej, próbując dotrzeć do ukrytych struktur determinujących sposoby użycia oprogramowania (zob. Fuller, 2003). Fuller został zresztą redaktorem „założycielskiej” publikacji promowanej przez Manovicha dyscypliny, *Software Studies. A Lexicon* (której fragmenty prezentujemy w tym numerze „Kultury Popularnej”), wobec czego objęcie wymienionych propozycji badawczych wspólnym szyldem programoznawstwa staje się faktem. Analizy Fullera w pewnym stopniu bliskie są zainteresowaniom Aleksandra R. Gallowaya, którego krytyka oprogramowania jest silnie zaangażowana politycznie – w najnowszej książce *The Exploit*, napisanej wspólnie z Eugenem Thackerem, wprost postawione zostaje pytanie: kto rządzi światem – prezydent USA czy nieżyjący już Jon Postel, współtwórca protokołów komunikacyjnych internetu? Politycznie zaangażowane wydanie studiów nad oprogramowaniem lokuje się w kręgu antyhumanizmu i posthumanizmu, choć równocześnie przypomina postulaty znane z brytyjskich studiów kulturowych, jak nawoływanie do wypracowania narzędzi umożliwiających krytykę opresyjnej sieci i prób przezwyciężenia jej kontroli – na poziomie edukacji, ale i działań bezpośrednich (przykładem tych drugich może być zrealizowany przez Gallowaya i nagrodzony na Ars Electronica w roku 2002 projekt *Carnivore* – program monitorujący przepływ danych w internecie, wykorzystujący kod stworzony dla potrzeb FBI; w ten sposób myślenia wpisują się też projekty akcyjnego ramienia Critical Art Ensemble – Electronic Disturbance Theater, które w roku 1997 uruchomiło akcję FloodNet (zob. Critical Art Ensemble, 2004). Galloway podkreśla problem depolityczacji dyskursu wokół nowych technologii: „Właściwie brak jakiegokolwiek ruchu intelektualnego zajmują-

cego się polityczną analizą algorytmów. Podobnie nie istnieje alternatywny ruch poświęcony tworzeniu i rozwojowi alternatywnych algorytmów »progresywnych« (2006: 319).

Programoznawstwo to także refleksja nad historycznymi przemianami oprogramowania – w ten nurt wpisują się teksty Henry’ego Lowogda, najnowsza książka Manovicha, czy prace Iana Bogosta i Nicka Montforta, którzy rozszerzają koncept *software studies* na poziom struktur sprzętowych, na których uruchamiane jest oprogramowanie. Ich *platform studies* zajmują się badaniem platformy technologicznej, która warunkuje uruchamiany na niej kod. W tej ostatniej wersji badań nad kulturą informatyczną wyraźnie pobrzmiewają echa eseju Friedricha Kittlera *There Is No Software*. Jego autor, przywoływany regularnie w artykułach z kręgu *software studies*, zasługuje zresztą na poświęcenie mu uwagi – bo studia nad oprogramowaniem czytane przez pryzmat Kittlera jawią się jako coś więcej, niż próba dokonania kolejnego, po wizualnym i performatywnym, zwrotu w humanistyce. W tym ujęciu stanowią nie tyle powrót do tekstu w jego informatycznym wydaniu, ale – na głębszym poziomie – stopniową akceptację technodeterminizmu. Kultura staje się problemem technologicznym.

Filozofia Kittlera jest tyleż radykalna, co wciąż słabo znana w Polsce (choć odwołania do niej pojawiają się między innymi w tekstach Andrzeja Gwoździa). Według niemieckiego myśliciela to nie – jak u McLuhana – maszyny są przedłużeniem ludzi, lecz raczej odwrotnie: ludzie stają się odbiciami swoich maszyn, odpowiadają na ich żądania. Ten zwrot dokonał się za sprawą komputerów, które w pewnym sensie doprowadziły do końca mediów czy wręcz końca humanistyki: „Świat, w którym żyliśmy przez ostatnie 40 lat, nie dzieli się już na kamienie, rośliny i zwierzęta, ale na nieświętą trójcę hardware’u, software’u i wetware’u” (Kittler 1999). Pisanie tekstów przez ludzi odeszło do przeszłości w chwili, gdy inżynierowie Intela stworzyli pierwszy mikroprocesor – teraz jest uwarunkowane przez struktury komputerów, a jego następcą jest naświetlanie struktur procesorów na płytkach krzemu. Kittler, który uważa, że aby móc wypowiadać się o kulturze, trzeba znać choćby jeden język programowania, sam programuje w kodzie maszynowym i deklaruje nieufność wobec wszelkich interfejsów graficznych. Tego podejścia nie podziela Lev Manovich, który wymienia jednak Kittlera wśród autorów, którzy na proponowany przez niego paradygmat wywarli istotny wpływ. Autorzy z kręgu studiów nad oprogramowaniem mogą się bowiem z Kittlerem nie zgadzać, jeśli chodzi o szczegóły (według Kittlera oprogramowanie jako takie nie istnieje, ukrywa jedynie funkcjonowanie sprzętu, który w znacznej części nie jest programowalny; z kolei większość autorów identyfikujących się ze studiami nad oprogramowaniem przyznaje temu ostatniemu wymiar materialny). Łączy ich jednak wizja połączenia studiów nad kulturą z informatyką – znaczące, że większość „programoznawców” ma kompetencje koderów (zob. interesujący tekst o pracach Manovicha – Zawojski, 2008). Dlatego uwzględniające różne perspektywy badanie oprogramowania nie wyczerpuje koncepcji *software studies* (jak już wspominałem, wciąż dość niespójnej).

Jednym z forsowanych przez Manovicha postulatów jest otwarcie nauk o kulturze na język informatyki jako źródło narzędzi analizy – na poziomie opisu, ale też narzędzi rozumianych jako oprogramowanie do badania kultury. Połączenie teorii z praktyką znajduje wyraz w idei analityki kulturowej (*cultural analytics*), która traktuje kulturę jako zbiór danych, możliwy do matematycznej i graficznej interpretacji. Te dane to obiekty medialne obecne w sieci, ale i ślady, które zostawiają po sobie, często tego nieświadomi, użytkownicy. Wykorzystanie w badaniach kultury wsparcia informatycznego – powszechnego choćby w analizach biznesowych – ma pomóc przewyciężyć bezradność badaczy wobec zalewu strumieniem danych z internetu i pomóc wychwycić w masie danych nowe trendy i zjawiska. Podstawowym atutem analityki kulturowej ma być właśnie możliwość analizy globalnej, bo – jak twierdzi Manovich – efekt skali jest klu-

czowy dla zrozumienia przemian związanych z rozwojem internetu (zob. Manovich, 2008a, 2008b).

Czy ta – trzeba to podkreślić – radykalna zmiana paradygmatu w studiach nad kulturą pozwoli nam ją lepiej zrozumieć? Wydaje się, że w chwili obecnej przedstawiciele studiów nad oprogramowaniem przede wszystkim wskazują na zaniedbywany obszar badań kulturoznawczych i dyskutują, jakich narzędzi użyć, by sporządzić jego mapę. Droga do celu – zrozumienia tego, co znajduje się po drugiej stronie lustra ekranu – jest jednak wciąż długa. Uznając jednak wartość podsuwanych przez programoznawstwo zagadnień badawczych, zapraszamy Czytelników do lektury bloku tekstów zajmujących się kulturową refleksją nad oprogramowaniem.

BIBLIOGRAFIA

- Bogost, I., Montfort, N. (2007). *New Media as Material Constraint. An Introduction to Platform Studies*; <http://www.bogost.com/downloads/Bogost%20Montfort%20HASTAC.pdf> (15 XII 2008).
- Castells, M. (2007). *Spółczesność sieci*, Warszawa: PWN.
- Critical Art Ensemble (2004). *Nomadyczna władza i opór kulturowy*, „Kultura Popularna” nr 4(10), s. 123–131.
- Fuller, M. (2003). *Behind the Blip*, New York: Autonomedia.
- Fuller, M. (red., 2008). *Software Studies. A Lexicon*, Cambridge, MA: MIT Press.
- Galloway, A.R. (2006). *Protocol*, „Theory, Culture, Society”, vol. 23, no. 2–3, s. 317–320.
- Galloway, A.R., Thacker, E. (2007). *The Exploit. A Theory of Networks*, Minneapolis: University of Minnesota Press.
- Kittler, F. (1995). *There is No Software*; <http://www.ctheory.net/articles.aspx?id=74> (15 XII 2008).
- Kittler, F. (1999). *On the Implementation of Knowledge – Toward a Theory of Hardware*; dostępny: <http://www.hydra.umn.edu/kittler/implement.html> (15 XII 2008).
- Lessig, L. (2005). *Wolna kultura*, Warszawa: WAIp; dostępny: <http://www.futrega.org/wk/> (15 XII 2008).
- Lowood, H. (2001). *The Hard Work of Software History*, „RBM: A Journal of Rare Books, Manuscripts, and Cultural Heritage” 2, no. 2, s. 141–61; dostępny: <http://www.ala.org/ala/mgrps/divs/acrl/publications/rbm/backissuesvol2no/lowood.PDF> (15 XII 2008).
- Manovich, L. (2006). *Język nowych mediów*, Warszawa: WAIp, tłum. P. Cypriański.
- Manovich, L. (2008a). *Cultural Analytics: Analysis and Visualization of Large Cultural Data Sets*; http://culturavis.com/CULTURAL%20ANALYTICS_2008.doc (15 XII 2008).
- Manovich, L. (2008b). *Scale Effects* (notatki z wykładu udostępniane przez autora); http://manovich.net/DOCS/scale_lecture_2008.doc (15 XII 2008).
- Manovich, L. (2008c). *Software Takes Command*; <http://softwarestudies.com/softbook> (15 XII 2008).
- Marino, M.C. (2006). *Critical Code Studies*; <http://www.electronicbookreview.com/thread/electropoetics/codology> (15 XII 2008).
- Varnelis, K. (red., 2008). *Networked Publics*, Cambridge, MA: MIT Press.
- Zawojski, P. (2008). *Soft Cinema Lva Manovicha i Andreasa Kratky’ego*. *Język nowych mediów w praktyce*, „Kwartalnik filmowy”, nr 60, s. 61–82; dostępny: <http://www.zawojski.com/2008/06/07/soft-cinema-lva-manovicha-i-andreasa-kratky’ego-jezyk-nowych-mediow-w-praktyce> (15 XII 2008).