

PAWEŁ STACEWICZ



FILOZOFICZNA REFLEKSJA NAD ISTOTĄ INFORMATYKI

Recenzja książki:

Izabela Bondecka-Krzykowska, *O przedmiocie badań informatyki. Studium filozoficzne*, Wydawnictwo Naukowe UAM, Poznań 2023

ABSTRACT. Paweł Stacewicz, *Filozoficzna refleksja nad istotą informatyki* [A philosophical reflection on the essence of computer science] edited by Sławomir Leciejewski, „Człowiek i Społeczeństwo” vol. LVIII: *Społeczny wymiar rewolucji informatycznej* [The social dimension of the information technology revolution], Poznań 2024, pp. 181–190, Adam Mickiewicz University. ISSN 0239-3271, <https://doi.org/10.14746/cis.2024.58.9>.

The paper is a review of Izabela Bondecka-Krzykowska's book entitled „O przedmiocie badań informatycznych. Studium filozoficzne” (eng. “On the subject of computer science research. A philosophical study”). It discusses the main theses of the book, the composition of the content in individual chapters and suggestions for topics that could be taken up in subsequent editions of the monograph. A number of polemical remarks were also formulated.

Keywords: philosophy of computer science, computer science methodology, computer, computer program, algorithm, information

Paweł Stacewicz, Politechnika Warszawska, Wydział Administracji i Nauk Społecznych, plac Politechniki 1, 00-661 Warszawa, e-mail: pawel.stacewicz@pw.edu.pl, <https://orcid.org/0000-0003-2500-4086>.

Książka Izabeli Bondeckiej-Krzykowskiej pod tytułem *O przedmiocie badań informatyki. Studium filozoficzne* (Wydawnictwo Naukowe UAM, 2023) podejmuje fundamentalną dla filozofii informatyki kwestię istoty informatyki, czyli określenia przedmiotu badań tej dyscypliny oraz jego istotnych

właściwości. O tym, że kwestia ta pozostaje wciąż otwarta, świadczą żywe wśród samych informatyków dyskusje dotyczące nie tylko zakresu badań, lecz nawet samej nazwy dyscypliny. W różnych środowiskach, z różnych powodów, związanych także z chęcią uwypuklenia takiego czy innego przedmiotu badań, używa się różnych określeń, przykładowo: wybijających na pierwszy plan pojęcie informacji – jak „informatyka” (pol.) czy „informatik” (niem.), podkreślających rolę komputerów – tu mamy dominujące w literaturze anglojęzycznej określenie „computer science”, a także kładących nacisk na przetwarzanie danych – jak „computing” (ang.) czy „data science” (ang.) (zob. Knuth, 1974).

W tytule recenzowanej książki widnieje polski termin „informatyka”, nie oznacza to jednak, że autorka zawęży przedmiot badań analizowanej dyscypliny do informacji czy danych. Traktuje go o wiele szerzej, dokonując syntetycznego przeglądu literatury. To sprawia, że jej książka ma duże walory dydaktyczne, zwłaszcza dla osób słabo zorientowanych w filozoficznych aspektach informatyki. Przede wszystkim jednak recenzowana monografia stanowi autorskie opracowanie naukowe, w którym autorka stawia, objaśnia i uzasadnia oryginalne naukowe tezy.

Główne tezy książki

Centralną tezę książki, rozpisaną na trzy kolejne rozdziały i uzasadnianą w ich treści, stanowi stwierdzenie, że przedmiot badań informatyki i jej istotę zarazem określają trzy powiązane ze sobą pojęcia: komputer, program (komputerowy) oraz informacja. Łącząc je w jednym syntetycznym określeniu, autorka używa terminu „system komputerowy”, który odnosi się do całości złożonej z komputera (hardware) i oprogramowania (software), służącej do efektywnego przetwarzania informacji. To właśnie systemy komputerowe miałyby stanowić właściwy (choć złożony) przedmiot zainteresowania informatyków – zarówno w kontekście badań podstawowych, jak i zastosowań. Rozwinięcie tej tezy – polegające na precyzowaniu wyżej wymienionych pojęć z pozycji historycznych, metodologicznych i filozoficznych – stanowią trzy kolejne rozdziały książki: 1. Komputer, 2. Program komputerowy, 3. Informacja.

Tytuł rozdziału 2 ujawnia kolejną tezę, bardziej szczegółową, która głosi, że spośród dwóch pojęć pokrewnych, które często wskazuje się jako pojęcia oddające istotę informatyki, tj. algorytmu i programu komputerowego, ważniejsze jest to drugie. Ma ono większą wagę, ponieważ stanowi

o metodologicznej niezależności informatyki względem matematyki (która na długo przed powstaniem tej pierwszej zajmowała się algorytmami), a ponadto wiąże się silniej niż pojęcie algorytmu z informatyczną praktyką (tj. programowaniem, weryfikacją programów, tworzeniem języków programowania etc.). Jest to teza niewątpliwie kontrowersyjna, ale autorka jej konsekwentnie broni – ukazując, przede wszystkim w rozdziale 2, całą paletę istotnych dla informatyków zagadnień, które wiążą się właśnie z programami komputerowymi, a nie z algorytmami. Są to, na przykład, zagadnienie kryteriów identyczności programów, sprawdzania ich poprawności czy testowania.

We wstępie do książki znajdujemy dość przekonującą argumentację na rzecz wspomnianej tezy, jednak nabrałaby ona niewątpliwie większej mocy, gdyby w części wprowadzającej rozdziału 2 (lub nawet w ramach osobnego podrozdziału) została przedstawiona dyskusja na temat relacji metodologicznych między programem i algorytmem. W szczególności można by tutaj rozwinąć tezę zarysowaną we wstępie do książki, że algorytmy mają status obiektów abstrakcyjnych i jako takie podlegają badaniom matematycznym, natomiast programy mają naturę dualną (mieszaną). Z jednej strony, jako „pochodzące” od algorytmów, są czymś abstrakcyjnym, a z drugiej – jako uruchamiane na fizycznych komputerach – są czymś fizycznym i podatnym na badania empiryczne.

Kolejna teza książki, przewijająca się przez jej wszystkie rozdziały, mówi, że dla skutecznego uprawiania informatyki nie wystarczą metody formalne, wobec czego informatyki nie można uznać za naukę czysto formalną (jak matematyka czy logika). W szczególności w podrozdziale 1.1.2 autorka przekonująco argumentuje, że formalne dowody poprawności systemów komputerowych nie przesądzają o ich niezawodności w praktyce, dotyczą bowiem teoretycznych projektów tych systemów, a nie ich fizycznych implementacji. Również w podrozdziale 2.2 pt. „Epistemologia” (w domyśle: „Epistemologia programów komputerowych”) znajdujemy obszerną analizę metod tworzenia programów komputerowych, która wykazuje, że bez zastosowania metod inżynierskich (choćby testów praktycznych) nie jest możliwe stworzenie programu zgodnego z zadaną specyfikacją. Nie wystarczą do tego formalne (podobne do matematycznych) metody dowodzenia poprawności programów.

Pozostając w tym samym kontekście i ponownie odwołując się do informatycznej praktyki (podrozdział 1.1.2), autorka przygląda się krytycznie typowemu dla informatyki rozróżnieniu *hardware/software*. Okazuje się ono nieostre – nie pozwala zatem oddzielić aspektów inżynierskich i formalnych

systemów komputerowych. Autorka przywołuje tutaj, między innymi, przykład rozwiązań sprzętowych typu *firmware*. Ponieważ pełnią one funkcję oprogramowania, muszą być kwalifikowane nieostro jako będące w pewnym stopniu *hardware*, a w pewnym stopniu *software*.

Ostatnia teza ogólna, bardzo dla książki charakterystyczna, stanowi właściwie zbiór konstatacji negatywnych, formułowanych w różnych miejscach książki, dotyczących nieadekwatności (słabiej: niewystarczalności) zastanych w literaturze definicji podstawowych pojęć informatyki (tj. komputera, programu i informacji). Aby ją uzasadnić, autorka analizuje wnikliwie różne podejścia do definiowania tychże pojęć, krytykując każde z nich. Píše na przykład (s. 50): „Wśród badaczy zajmujących się zagadnieniami filozoficznymi związanymi z informatyką nie ma zgody co do natury urządzeń nazywanych komputerami; nie udało się nawet stworzyć powszechnie przyjmowanej definicji komputera”. W innych miejscach książki formułuje podobne diagnozy, wynikające ze starannego przeglądu różnych stanowisk w zakresie metodologii informatyki. Tego rodzaju szeroki, a w przypadku pojęcia informacji wręcz interdyscyplinarny, przegląd stanowi o wartości książki. Byłaby ona jednak znacznie wyższa, gdyby autorka nie poprzestała na krytyce i polemice, lecz przedstawiała częściej własne, oryginalne propozycje.

Szczególny niedosyt w tym względzie pozostawia lektura rozdziału 3, w którym omówiono syntetycznie różne teorie informacji, wykraczając znacznie poza obszar pojęciowy informatyki (np. w kierunku filozoficznego podejścia Luciana Floridiego). Gdyby autorka bardziej skoncentrowała się na samej informatyce, to być może doszłaby do wniosku, że informatycy zajmują się nie „informacją w ogóle”, lecz pewną szczególną jej formą, zwaną niekiedy danymi. Uwidacznia to chociażby terminologia branżowa, a w niej takie określenia jak „przetwarzanie danych” (ang. *data processing*) czy „baza danych” (ang. *database*). Uważam, że właściwy przedmiot zainteresowania informatyków stanowi informacja zredukowana do danych, a więc informacja zakodowana, a także ustrukturyzowana¹ w sposób pozwalający ją przetwarzać za pomocą komputerów (por. Stacewicz, 2017; Marciszewski i Stacewicz, 2011).

¹ W językach programowania wysokiego poziomu mamy do dyspozycji bardzo różne struktury danych, takie jak tablice, stopy czy drzewa. Wybór odpowiedniej struktury danych pozwala istotnie zwiększyć szybkość rozwiązywania problemu za pomocą programu komputerowego. Zob. Wirth, 1989.

Układ treści

Omawiana książka składa się z trzech rozdziałów głównych, obszernego wstępu i zakończenia. Każdy z rozdziałów kończy się syntetycznym podsumowaniem, w którym autorka streszcza, zazwyczaj na kilku stronach, główne punkty swoich analiz i najważniejsze tezy. Jest to bardzo cenny element monografii. Do książki dołączono obszerny, ponad 30-stronicowy dodatek pt. „Uwagi na temat ontologii rzeczywistości wirtualnej”.

Tytuły kolejnych rozdziałów – „Komputer”, „Program komputerowy” oraz „Informacja” – odzwierciedlają główną tezę książki, że istotę informatyki oddaje dobrze pojęciowa triada (komputer, program, informacja). Chociaż tworzące ją pojęcia są ze sobą nierozzerwalnie związane (bo komputery służą do przetwarzania informacji na podstawie programów), to każde z nich wyznacza inne obszary informatycznych badań i filozoficznej refleksji nad nimi. Przykładowo: pojęcie *komputer* prowadzi do pytań o cechy odróżniające komputery od innego rodzaju urządzeń technicznych, a pojęcie programu – do pytań o kryteria identyczności programów (np. czy dla stwierdzenia ich identyczności wystarczy, aby realizowały one ten sam algorytm?).

W rozdziale 1 zostały omówione różne podejścia do definiowania komputerów, osadzone w gruncie rzeczy w różnych paradygmatach uprawiania informatyki. W każdym z tych podejść przyjmuje się inną nadrzędną dla pojęcia *komputer* kategorię: artefaktów technicznych, maszyn abstrakcyjnych bądź maszyn programowalnych. Najwięcej miejsca poświęcono analizie różnych koncepcji artefaktów, tj. wytworów intencjonalnej działalności człowieka, służących realizacji określonych celów. W konkluzji swoich omówień i analiz autorka nie opowiada się wyraźnie za żadnym z podejść; poprzestaje na przeglądzie różnych stanowisk i uwagach krytycznych².

Rozdział 2 przybliży kluczowe dla stanowiska autorki pojęcie *program komputerowy* (przypomnijmy: program stanowi dla informatyków pojęcie centralne; programy są też dla informatyków ważniejsze niż algorytmy).

² W ramach każdego z podejść autorka uwypukla stwierdzenie, że komputer jest maszyną uniwersalną (i programowalną). W tym kontekście warto zauważyć, że w przypadku realnych komputerów można mówić o ograniczonej uniwersalności, ograniczonej np. własnościami procesora i rozmiarami dostępnej pamięci. W przypadku komputerów ujętych abstrakcyjnie, np. jako maszyny Turinga, uniwersalność także jest ograniczona, ponieważ istnieją matematyczne dowody nierozwiązywalności (praktycznej lub zasadniczej) problemów za pomocą tego rodzaju maszyn. Por. Harel, 2000.

Rozdział dzieli się na część ontologiczną, dotyczącą statusu bytowego programów komputerowych (np. czym różni się sam program od jego implementacji), i część epistemologiczną, dotyczącą głównie kryteriów i metod badania poprawności programów³. Osią tematyczną części pierwszej jest zagadnienie relacji ontologicznych między trzema postaciami programów komputerowych: (1) postacią abstrakcyjną, która manifestuje się najlepiej w algorytmie realizacji pewnego zadania oraz formalnej specyfikacji tego zadania, (2) postacią konkretną i językową zarazem, którą stanowi konkretny tekst programu (zapisany przy użyciu instrukcji określonego języka programowania), oraz (3) postacią konkretną fizyczną, której odpowiada wykonanie programu, czyli pewien proces fizyczny (polegający, na przykład, na przepływie sygnałów elektrycznych wewnątrz maszyn cyfrowych). Autorka bada te relacje w dwóch kontekstach: tworzenia programów (od specyfikacji do implementacji) oraz struktury i działania programów (jak to się dzieje, że pewna określona struktura przekłada się na konkretne działanie?). Ciekawym wynikiem tej części jest teza, że programy komputerowe nie mogą być obiektami *stricte* matematycznymi, ponieważ w warstwie zapisu formalnego są albo zdaniami syntetycznymi, albo zdaniami nie-analitycznymi innego rodzaju. Dzieje się tak wskutek używania w nich instrukcji podstawienia.

W drugiej części rozdziału, epistemologicznej, autorka podejmuje właściwie jedno złożone zagadnienie: pod jakimi warunkami i na mocy jakich metod program można uznać za poprawny? I tutaj właśnie rozwija tezę, że dla sprawdzania poprawności nie wystarczą metody formalne, pokrewne dowodom matematycznym, lecz należy je co najmniej uzupełnić o metody empiryczne (empiryczne testy poprawności). Uważam, że w części tej powinno być podjęte inne jeszcze zagadnienie, dotyczące *stricte* formalnych ograniczeń metody algorytmicznej, polegających na tym, że dla pewnych klas programów (np. programów dla maszyn cyfrowych) istnieją problemy nierozwiązywalne (zob. Harel, 2000). Zwracam na to uwagę w ostatniej części recenzji.

Zawartość ostatniego rozdziału, zatytułowanego „Informacja”, zdecydowanie wykracza poza przedmiot badań informatyki – co oczywiście znajduje uzasadnienie w filozoficznym charakterze publikacji. Znajdujemy w nim szeroką prezentację różnych, w gruncie rzeczy interdyscyplinarnych,

³ *Notabene* podział na część ontologiczną i epistemologiczną rozważań mógłby być stosowany konsekwentnie w pozostałych rozdziałach, dotyczących odpowiednio komputera i informacji. Przykładowo: pytanie „czym jest informacja?” ma charakter ontologiczny, zaś pytania o różnice między informacją a wiedzą mają charakter epistemologiczny.

teorii informacji: począwszy od teorii ilościowej Shannona, przez teorię algorytmiczną Chaitina, a skończywszy na filozoficznym ujęciu Floridiego. Prezentację tych teorii autorka wzbogaca o szereg uwag krytycznych, dotyczących na przykład ujęć wyłącznie ilościowych (w stylu Shannona). Ta część książki musi wywołać u czytelnika silne przekonanie, że pojęcie informacji, choć tak często używane w analizach i dyskusjach naukowych, wciąż dalekie jest od jakiegś jednoznacznej i podzielanej przez wszystkich definicji. Pewną słabością analiz autorki jest brak wyodrębnienia tych wątków ogólnej teorii informacji, które mają bezpośrednie znaczenie dla informatyki. Spostrzeżenia na ten temat znajdujemy w podsumowaniu rozdziału, a nie w głównej jego części.

Uzupełnieniem głównych rozdziałów książki jest obszerny, liczący ponad 30 stron, Dodatek, poświęcony ontologii rzeczywistości wirtualnej. Jego najbardziej rozbudowana część dotyczy identyfikacji cech charakterystycznych rzeczywistości wirtualnej, takich jak interaktywność i immersja. W częściach kolejnych opisano próby (pojęciowego) odróżniania obiektów wirtualnych od obiektów realnych, fikcyjnych i symulacji. Nie jest do końca jasne, jaką funkcję w książce pełni Dodatek. Czy chodzi o uwypuklenie pewnych kwestii filozoficznych z zakresu ontologii, które tylko szczątkowo zostały poruszone w głównej części książki (a które wnoszą coś nowego do rozważań filozofów o naturze przedmiotów intencjonalnych, artefaktów etc.)? Czy może jego celem jest zwrócenie uwagi czytelnika na szczególnego rodzaju systemy komputerowe, zdolne do kreowania rzeczywistości wirtualnej? W tym drugim ujęciu Dodatek stanowiłby swoiste studium przypadku: wskazujemy szczególnie przedmiot zainteresowania informatyków, określamy jego istotne właściwości i formułujemy związane z nim specyficzne kwestie filozoficzne. Uważam, że to byłoby najlepsze rozwiązanie i gdyby zostało ono przyjęte, to po pewnych modyfikacjach jego zawartość można by włączyć do rozdziału dotyczącego programów komputerowych.

Tematy nieobecne w książce, a warte podjęcia w przyszłości

Podsumowując powyższe uwagi (również polemiczne), muszę stwierdzić, że książka Izabeli Bondeckiej-Krzykowskiej jest bardzo dobrze opracowanym i pobudzającym do refleksji źródłem wiedzy z zakresu metodologii i filozofii informatyki. Mimo to uważam, że w jej kolejnych wydaniach – na co z pewnością zasługuje i temat, i sposób jego prezentacji – warto byłoby uwzględnić pewne zagadnienia, które są dla informatyki istotne, a nie zostały

przez autorkę w ogóle lub w dostatecznym stopniu omówione. Opiszę krótko cztery takie zagadnienia.

Główny nurt informatyki a nurty poboczne. Analizy przedstawione w książce koncentrują się na głównym nurcie badań informatycznych, związanym z komputerami cyfrowymi i cyfrowymi technikami przetwarzania danych. Tymczasem informatyka w sensie szerszym zajmuje się również technikami innego typu, na przykład analogowymi czy kwantowymi, które mają swoje teoretyczne podstawy w niestandardowych modelach obliczeń (wymagają też innych metod programowania). Na szczególną uwagę zasługują tu komputery i obliczenia analogowe ciągłe, które dominowały do lat 40. XX wieku, a dziś ponownie budzą spore zainteresowanie, zwłaszcza w kontekście prób informatycznego modelowania aktywności nerwowej mózgu (zob. Maley, 2011; Stacewicz, 2020).

Trzeba przyznać, że autorka kwestię obliczeń analogowych sygnalizuje w podrozdziale 1.1. Zastanawia się mianowicie, czy wymóg cyfrowości (tu warto wyjaśnić dodatkowo, że chodzi o dyskretną, tj. nieciągłą, naturę elementarnych operacji i przetwarzanych danych) jest konieczny, aby maszynę uznać za komputer. Kwestię tę zdaje się pozostawiać otwartą, przywołując właśnie przykłady komputerów analogowych. W dalszej części rozdziału 1 pisze już jednak wyłącznie o maszynach cyfrowych, nie wracając do koncepcji maszyn przetwarzających sygnały ciągłe (analogowych). Nawet nadmieniając o różnych abstrakcyjnych czy matematycznych modelach komputerów, nie wymienia z nazwy, a tym bardziej nie opisuje, żadnego modelu poza uniwersalną maszyną Turinga (która stanowi model komputerów cyfrowych).

W obecnej wersji monografii nie znajdziemy również żadnych informacji i analiz dotyczących obliczeń kwantowych czy biologicznych (inspirowanych biologicznie lub realizowanych z wykorzystaniem materii organicznej), a tymczasem są to dość rozbudowane i przyszłościowe obszary badawcze współczesnej informatyki. Pewnym rozwiązaniem na przyszłość byłaby rezygnacja z dodatku poświęconego rzeczywistości wirtualnej i zastąpienie go rozdziałem (lub podrozdziałem) na temat niecyfrowych technik obliczeniowych.

Historia idei programistycznych. Rozdział 1 monografii zaczyna się od bardzo ciekawych i potrzebnych, moim zdaniem, uwag o historii komputerów⁴. Zarówno ze względów merytorycznych, jak i redakcyjnych

⁴ Ze względu na silnie filozoficzny charakter książki powinny się tam znaleźć informacje o arytmometrach zaprojektowanych i skonstruowanych w XVII wieku przez dwóch

(zapewnienie pewnej „symetrii” w zawartości rozdziałów), na początku rozdziału 2 warto byłoby omówić krótko historię istotnych dla rozwoju informatyki idei programistycznych – takich jak koncepcja programu składowanego w pamięci (wprowadzona wprawdzie przez von Neumanna, ale obecna teoretycznie już w modelu uniwersalnej maszyny Turinga), koncepcja wielopoziomowości języków programowania, czy bardziej już współczesna koncepcja programów uczących się (wykorzystywana intensywnie przy projektowaniu systemów sztucznej inteligencji).

Warto pokusić się też o rozwinięcie tematu języków programowania, w tym o zarysowanie ich typologii – rozróżniając, na przykład, języki proceduralne i deklaratywne. Jest to uzasadnione postawioną w książce tezą, że to program właśnie (a nie algorytm) stanowi pojęcie centralne, (współ)definiujące informatykę. Temat ewolucji języków programowania warto byłoby podjąć w sugerowanym wyżej wprowadzeniu historycznym do rozdziału 2.

Kwestia złożoności obliczeniowej programów komputerowych.

W rozdziale 2, w jego części epistemologicznej, nie znajdujemy żadnych odniesień do złożoności obliczeniowej programów komputerowych. Jest to tymczasem ich istotna właściwość, różnicująca programy o tym samym celu działania. Ma ona dodatkowo ogromne znaczenie dla epistemologii – może bowiem wskazywać na ograniczenia algorytmicznej metody rozwiązywania problemów, czyli pewnej metody zdobywania wiedzy. Mam na myśli fakt, że problemy, dla których nie istnieją programy o dostatecznie niskiej złożoności, są praktycznie nierozwiązywalne metodą algorytmiczną (zob. Harel, 2000).

Sztuczna inteligencja. Wielką nieobecną w obecnej wersji książki jest sztuczna inteligencja, czyli ta dziedzina informatyki, która zajmuje się algorytmizacją i automatyzacją procesów przypominających procesy poznawcze człowieka (jak wnioskowanie czy uczenie się) (zob. Stacewicz, 2010). Systemy sztucznej inteligencji stanowią z pewnością jeden z głównych przedmiotów badań informatyki współczesnej (zob. Russel i Norvig, 2020), a pewne ich typologie (np. systemy logicystyczne *versus* naturalistyczne), jak również cechy szczególne (np. zdolność do uczenia się), są zdecydowanie warte omówienia w przyszłości. Sugeruję, aby w kolejnej wersji książki

filozofów: Pascala i Leibniza. W przypadku Leibniza warto podkreślić także inne jego dokonania, niezwykle istotne dla dalszego rozwoju maszyn obliczeniowych: wynalazek binarnej notacji liczb, projekt maszyny binarnej oraz śmiałą wizję maszyn zdolnych rozwiązywać problemy (nie tylko matematyczne) za pomocą obliczeń. Zob. Trzęsicki, 2006.

dodać nowy rozdział pt. „Pewne szczególne przedmioty badań informatycznych” i w jego ramach omówić zarówno systemy rzeczywistości wirtualnej, jak i systemy sztucznej inteligencji.

Literatura

- Harel, D. (2000). *Rzecz o istocie informatyki. Algorytmika*. Warszawa: Wydawnictwa Naukowo-Techniczne.
- Knuth, D.E. (1974). Computer Science and its Relation to Mathematics. *American Mathematical Monthly*, 81(4), 323–343.
- Maley, C.J. (2011). Analog and Digital, Continuous and Discrete. *Philosophical Studies*, 155(1), 117–131.
- Marciszewski, W., Stacewicz, P. (2011). *Umysł – Komputer – Świat. O zagadce umysłu z informatycznego punktu widzenia*. Warszawa: EXIT.
- Russel, S., Norvig, P. (2020), *Artificial Intelligence: A Modern Approach*. Berkeley: Prentice Hall.
- Stacewicz, P. (2010), *Umysł a modele maszyn uczących się. Współczesne badania informatyczne w oczach filozofa*. Warszawa: EXIT.
- Stacewicz, P. (2017), *O redukcji informacji do danych*. W: P. Stacewicz, *Różne oblicza informacji* (ss. 11–21). Warszawa: Oficyna Wydawnicza Politechniki Warszawskiej.
- Stacewicz, P. (2020). Analogicity in Computer Science. Methodological Analysis. *Studies in Logic, Grammar and Rhetoric*, 63, 69–86.
- Trzęsicki, K. (2006). Leibnizjańskie inspiracje informatyki. *Filozofia Nauki*, 55, 21–48.
- Wirth, N. (1989). *Algorytmy + struktury danych = programy*. Warszawa: Wydawnictwa Naukowo-Techniczne.