



**ALEKSANDER PIECUCH**

## **Zaniedbana algebra a nauczanie informatyki**

---

### **Neglected Algebra and Informatics Teaching**

Doktor habilitowany profesor UR, Uniwersytet Rzeszowski, Wydział Matematyczno-Przyrodniczy, Katedra Inżynierii Komputerowej; Laboratorium Zagadnień Społeczeństwa Informatycznego, Polska

#### **Streszczenie**

Artykuł podejmuje próbę zwrócenia uwagi na zagadnienia związane z nauczaniem logiki w ramach przedmiotu informatyka. Jeszcze początkiem lat 80. XX w. była ona obecna w programach nauczania. W świetle zapowiadanych przez Ministerstwo Edukacji Narodowej reform związanych z kształceniem informatycznym zagadnienia logiki nabierają szczególnego znaczenia. Nauczanie programowania sprofilowanego na np. programowanie robotów, względnie innych platform programowalnych, bez znajomości algebry Boole’a nie będzie dobrym rozwiązaniem z punktu widzenia kompetencji uczniów.

**Słowa kluczowe:** logika, algebra Boole’a, nauczanie informatyki, programowanie

#### **Abstract**

The article attempts to draw attention to issues related to the teaching of logic within the subject of computer science. Even in the early 1980s, it was present in the curriculum. In light of the IT reforms announced by MEN, logic issues are of particular importance. Teaching programming such as programming robots or other programming platforms without the knowledge of Boole algebra will not be a good solution for students competence.

**Keywords:** logic, Boolean algebra, informatics teaching, programming

---

#### **Wstęp**

Historia logiki, na której opiera się współczesna informatyka, sięga swoimi początkami do XVII w. Już wówczas Leibnitz (1646–1716) zastanawiał się nad stworzeniem takiego języka, który pozwoliłby wszystko zredukować do prostego stwierdzenia: że coś jest prawdą lub fałszem. Można zatem mówić o pierwszych próbach algebraizacji logiki (Marciszewski, 1988). Właśnie jemu zawdzięczamy stosowany do dziś system binarny (opisany w artykule *Explication de l'arithmétique binaire* w 1703 r.) (Kofler, 1956), chociaż prawdopodobnie jako filozofowi bardziej chodziło mu o formułowanie i rozstrzygnięcie problemów natury filozoficznej.

Reguły matematyczne analizy zdań logicznych zostały opracowane w 1854 r. przez angielskiego matematyka Boole'a. Dodajmy, że przedtem logika była mało przejrzysta i trudna w zastosowaniach. Istota przekształceń zaproponowana przez Boole'a wywodzi się z koncepcji Leibniza, że zdanie logiczne może być wyłącznie określone jako prawdziwe lub fałszywe. Inaczej rzecz ujmując – ma się do czynienia z tzw. logiką dwustanową, gdzie zdaniu prawdziwemu można przypisać wartość logiczną równą 1, natomiast zdaniu fałszywemu wartość logiczną równą 0. System, który stworzył Boole od jego nazwiska nazywany jest algebrą Boole'a (Kaula, 2011). Podstawowe prawa przekształceń logicznych w algebrze boolowskiej opisuje pięć aksjomatów (praw): łączności, przemienności, rozdzielczości, de Morgana<sup>1</sup>, tożsamościowe. Praktycznego wykorzystania algebra doczekała się dopiero w 1938 r., kiedy to Shannon wykorzystał ją do opisu układów przekaźnikowych, a te, jak wiadomo, charakteryzuje jeden z dwóch stanów: włączenia lub wyłączenia, co może odpowiadać zapisowi w postaci „1” lub „0” logicznego<sup>2</sup>.

Dominujący okres, w którym przekaźniki odgrywały kluczową rolę we wszelakiego rodzaju konstrukcjach, mamy już za sobą (np. komputery konstruowane przez Zusa). Niemniej stosuje się je nadal z dużym powodzeniem głównie w rozwiązaniach z zakresu automatyki. Jednak w większości współczesnych konstrukcji ich funkcje przejęły elektroniczne układy cyfrowe, w których stany wejściowe i wyjściowe – inaczej dane wejściowe i dane wyjściowe – opisuje się za pomocą jednej z dwóch wartości „0” lub „1” (analogicznie jak w przekaźnikach: włączony, wyłączony)<sup>3</sup>. Stąd „1” lub „0” logiczne jest najmniejszą porcją informacji reprezentowaną w systemie dwójkowym (binarnym), którą przyjęto nazywać bitem od słów *binarydigit*.

## Dlaczego technika cyfrowa?

Wszelakiego rodzaju urządzenia techniczne, którymi otaczamy się na co dzień, w większości są już urządzeniami skonstruowanymi na bazie techniki cyfrowej. Zaliczamy do nich oczywiście komputery, smartfony, telewizory, ale także inne sprzęty z rodziny AGD, np. pralki automatyczne, w których programatory elektromechaniczne zastąpiono programatorami elektronicznymi. Technika cyfrowa nie jest więc wizją przyszłości, ale rzeczywistością techniczną,

---

<sup>1</sup> August De Morgan (27 czerwca 1806 r. – 18 marca 1871 r.) – angielski matematyk i logik. Wniósł do rozwoju logiki prawa odnoszące się do negacji koniunkcji i negacji alternatywy. Sformułowane przez niego prawa znane są pod nazwą praw De Morgana.

<sup>2</sup> Sposób, w jaki przypisane są wartości logiczne do wartości napięć, zależy od przyjętej konwencji, stąd rozróżnia się tzw. logikę dodatnią i logikę ujemną

<sup>3</sup> To bardzo duże uproszczenie. W rzeczywistości ma się do czynienia z pewnymi przedziałami napięć odpowiadającymi odpowiednio „0” i „1” logicznej. Przyjmując ten opis za zgodny z rzeczywistością, precyzyjniejsze byłyby stosowanie określenia dla „0” logicznego – *poziom niski*, a dla „1” logicznej – *poziom wysoki*.

którą się otaczamy i w której funkcjonujemy. Ten cyfrowy świat opisuje się, korzystając z podstawowych założeń algebry Boole'a. Wypada w tym miejscu dowieść wyższości technologii cyfrowych nad technologiami analogowymi.

Cechą charakterystyczną dla sygnałów analogowych jest ich mała dokładność wynikająca z tzw. szumów elektrycznych. Znane są i stosowane w elektronice metody ich ograniczania, ale niestety nie całkowitej eliminacji. Ostatecznie ten czynnik decyduje o „jakości” sygnału analogowego. Z tego punktu widzenia informacja, którą przesyła się za pomocą sygnału analogowego, może zostać zniekształcona. Tych wad pozbawiona jest technika cyfrowa. Nie oznacza to, że w układach cyfrowych nie występują szumy, ale nie mają one większego znaczenia<sup>4</sup>, bowiem przesyłane są tylko dwie wartości odpowiadające „0” lub „1” logicznej. Nie występuje zatem problem zniekształcenia informacji. Dzięki temu uzyskuje się możliwość:

- uproszczenia układów elektronicznych,
- niezakłóconego przesyłu informacji (odporność na zakłócenia),
- transmitowania sygnału na duże odległości bez utraty jakości,
- gromadzenia praktycznie nieograniczonej ilości informacji,
- kodowania/dekodowania sygnałów analogowych z wykorzystaniem przetworników A/C i C/A,
- nadawania w trakcie programowania przez programistę określonych funkcjonalności układom elektronicznym.

Współczesna elektronika cyfrowa zatem w całości bazuje na układach logicznych, które to wykonują wszystkie operacje na podstawie algebry Boole'a.

### **Nauczanie informatyki**

Nauczanie informatyki w polskich szkołach trwa już nieprzerwanie od ponad 30 lat. W tym czasie wielokrotnie zdążyły zmienić się już programy nauczania, które dostosowywano do zmian technologicznych, do zmieniającego się oprogramowania i potrzeb użytkowników. Warto jednak zwrócić uwagę, że nacisk w kształceniu informatycznym został położony w rzeczywistości na technologii informacyjne. Miały one za zadanie przygotować młode pokolenie do sprawnego i efektywnego wykorzystywania komputerów w codziennym życiu. W funkcjonujących programach nauczania trudno zatem odnaleźć problematykę, która sięgałaby do samych źródeł informatyki i dawała podstawy do pełnego zrozumienia zasad funkcjonowania współczesnych komputerów. Można polemizować, czy ta wiedza jest bezpośrednio użyteczna. Oczywiście, że nie jest, zresztą analogicznie jak w przypadku pozostałych przedmiotów kształcenia, ale

---

<sup>4</sup> W technice cyfrowej także ogranicza się do minimum występowanie szumów. Podstawowy warunek, który musi zostać spełniony, to: wartość szumu nie może być większa od różnicy napięć pomiędzy stanem „0” logicznego i „1” logicznej.

z pewnością daje szersze spojrzenie na otaczający nas świat, jego problemy, na prawa i procesy, z którymi spotykamy się codziennie, a są one w większości przypadków nieuświadomione.

Do tej kategorii problemów można zaliczyć elementy logiki, a w szczególności algebrę Boole'a, która stanowi podstawę funkcjonowania współczesnych komputerów. To prawda, że w programach nauczania odnajdziemy elementy architektury komputerów, metody kodowania informacji (w tym system binarny), niemniej występują one w bardzo okrojonej formie. Jeszcze w latach 70. i w pierwszej dekadzie lat 80. XX w. elementy logiki występowały w kształceniu informatycznym na poziomie szkoły średniej (zob. Piecuch, 2008). Obecnie nie znajdziemy ich nawet w programach kształcenia matematycznego.

Abstrahując od zagadnień typowych dla informatyki, wydaje się, że chyba błędnie postrzegamy logikę tylko w jej kontekście. Jak zauważa Kant (2009, s. 87), „[l]ogikę ogólną zbudowano według planu bardzo dokładnie zgadzającego się z podziałem władz poznawczych. A są nimi: intelekt, rozum i rozum. Stąd nauka ta zajmuje się w swojej analityce pojęciami, sądami i wnioskami zupełnie zgodnie z czynnościami i kolejnością tych sił umysłu, które obejmujemy rozległą nazwą rozumu w ogóle”. W innym zaś miejscu dopowiada: „Ogólna lecz czysta logika (...) stanowi kanon intelektu i rozumu, ale jedynie, co do formalnych aspektów jego używania” (Kant, 2009, s. 28). Wnioskować zatem należy, że jest ona przydatna nie tylko w ściśle określonych przypadkach matematycznych i/lub informatycznych, lecz także w życiowych, kiedy trzeba wyprowadzać wnioski, a na ich podstawie podejmować różnorakie decyzje. Z filozoficznego punktu widzenia „logika jest to nauka o naturze, sposobach i wartości uzasadnienia” (Tkaczyk, 2017, s. 7). Jeśli spojrzeć na logikę z prakseologicznego punktu widzenia, dochodzimy do wniosku, że celem jej poznania jest „umiejętność przestrzegania umów terminologicznych, umiejętność określenia struktury logicznej wypowiedzi, umiejętność sprawdzania tautologiczności formuł logiki pierwszego rzędu, definiowanie jednych terminów za pomocą drugich, precyzyjne formułowanie poglądów, odróżnianie zdań uzasadnionych od nieuzasadnionych i umiejętność przeprowadzenia analizy dowolnej argumentacji” (Olszewski, 2017, s. 45). Pomijając już inne aspekty, chociażby z tych względów, na które wskazują cytowani autorzy, logika godna jest zainteresowania.

Dla naszych potrzeb odwołam się jednak do definicji logiki matematycznej. W tym rozumieniu logika jest „dziedziną matematyki zajmującą się badaniem własności wnioskowania matematycznego oraz modeli teorii matematycznych” (<http://sjp.pl/logika> 2017). Sama definicja, chociaż umiejscawia logikę w obszarze nauk matematycznych, ma wręcz elementarne znaczenie i dla informatyki. Po pierwsze, dlatego, że istota funkcjonowania komputera zasadza się na systemie binarnym. Po drugie, każda operacja logiczna wykonywana jest na podstawie algebry Boole'a. W kontekście zapowiadanych zmian w systemie

kształcenia informatycznego znajomość omawianej algebry nabiera jeszcze innego i bardziej doniosłego znaczenia. Mam tu na myśli zwrot w kierunku nauczania programowania na wszystkich szczeblach kształcenia (szkoła podstawowa, szkoła ponadpodstawowa – zgodnie z zapowiedziami).

Znamy już podstawę programową kształcenia ogólnego (PPKO) dla przedmiotu informatyka dla szkół podstawowych. Zwróćmy uwagę na położony akcent na umiejętności związane z programowaniem. Podstawa programowa przywołuje je w wielu miejscach, wskazując na: „4) kreatywne rozwiązywanie problemów z różnych dziedzin ze świadomym wykorzystaniem metod i narzędzi wywodzących się z informatyki, w tym programowanie” (Rozporządzenie, 2017). W innych miejscach PPKO osiągnięcia uczniów w tym zakresie wyrażono w sformułowaniach: „1) programuje wizualnie: proste sytuacje lub historyjki według pomysłów własnych i pomysłów opracowanych wspólnie z innymi uczniami, pojedyncze polecenia, a także ich sekwencje sterujące obiektem na ekranie komputera bądź innego urządzenia cyfrowego (kl. I–III)” (Rozporządzenie, 2017); „c) sterowanie robotem lub obiektem na ekranie; (kl. IV–VI)” (Rozporządzenie, 2017); „1) projektuje, tworzy i zapisuje w wizualnym języku programowania: a) pomysły historyjek i rozwiązania problemów, w tym proste algorytmy z wykorzystaniem poleceń sekwencyjnych, warunkowych i iteracyjnych oraz zdarzeń, b) prosty program sterujący robotem lub innym obiektem na ekranie komputera (kl. IV–VI)” (Rozporządzenie, 2017); „1) projektuje, tworzy i testuje programy w procesie rozwiązywania problemów. W programach stosuje: instrukcje wejścia/wyjścia, wyrażenia arytmetyczne i logiczne, instrukcje warunkowe, instrukcje iteracyjne, funkcje oraz zmienne i tablice. W szczególności programuje algorytmy z działu I pkt 2; 2) projektuje, tworzy i testuje oprogramowanie sterujące robotem lub innym obiektem na ekranie lub w rzeczywistości (kl. VII–VIII)” (Rozporządzenie, 2017).

Z powyższych sformułowań wyraźnie wynika, że zwracamy się w kierunku bardziej utylitarnej nauki programowania. Bo jak inaczej rozumieć konstruowanie programu sterującego robotem? Obawy natomiast budzi brak konkretnych wskazań co do sposobu realizacji wspomnianej problematyki (zob. Piecuch, 2016b). Jak do tej pory nie pojawiły się informacje na temat realizacji przedmiotu informatyka w szkołach ponadpodstawowych. Z dużym prawdopodobieństwem należy przypuszczać, że ścieżka kształcenia informatycznego sprofilowana będzie również na nauczanie programowania. Jeśli chcemy wyjść poza schematyzm w kształceniu informatycznym i wyposażyć uczniów faktycznie w wiedzę i umiejętności praktyczne, konieczne jest zaproponowanie czegoś więcej. „Uczenie programowania dla samego nauczania – mija się z celem. Konstruowanie programu na rozwiązywanie równania liniowego czy kwadratowego nie zainteresuje uczniów, a jedynie zniechęci ich do dalszej nauki. Analogicznie jak nauczanie w szkole podstawowej i tu uczniowie powinni widzieć użyteczne

wyniki własnych zmagañ z programowaniem i to takie, które mają bezpośrednie przełożenie na praktykę dnia codziennego. Aktualnie przyszłość w rozwoju społeczno-gospodarczym leży w: elektronice, automatyce i robotyce. W takim kierunku powinna pójść nauka programowania” (Piecuch, 2016a, s. 33).

Jeśli hipotetycznie założyć właśnie taki model kształcenia, który w efekcie powinien dać uczniowi kompetencje przydatne również w sferze pozaszkolnej i przygotować w pewnym stopniu do funkcjonowania na rynku pracy, konieczne jest wprowadzenie dodatkowych treści do kanonu kształcenia. Tymi treściami są z całą pewnością zagadnienia z logiki, a ściślej mówiąc – jest to algebra Boole’a. Zakres przekazywanych treści powinien być oczywiście przedmiotem dyskusji. Przykładowo, niekoniecznie musi obejmować rachunek kwantyfikatorów, ale rachunek zdań jest już jak najbardziej pożądany.

Tylko rzeczywiste problemy teoretyczne przekładające się bezpośrednio na praktykę mają w efekcie siłę sprawczą, stąd jako takie powinny zostać włączone do programów nauczania informatyki. Umiejętność konstruowania układów logicznych o większej złożoności z wykorzystaniem podstawowych bramek (*gates*)<sup>5</sup> realizujących elementarne funkcje logiczne, wyznaczanie tablic prawdy (czasami zwanych tablicami zero-jedynkowymi) dla układów logicznych to konkretna wiedza i umiejętność pozwalająca spojrzeć zupełnie z innej perspektywy na układy logiczne i programowalne. Wspomniane kompetencje uczeń ma możliwość wykorzystać w swojej działalności pozaszkolnej, konstruując własne układy elektroniczne bądź używając do tego celu dostępnych na rynku gotowych modułów (zob. Piecuch, 2017), nadając im w trakcie programowania użyteczne z jego punktu widzenia pożądane funkcjonalności.

Bez wiedzy z zakresu algebry Boole’a napisanie programu mającego znamiona użyteczności na dowolną platformę mikroprocesorową jest bardzo trudne. Niewystarczająca może się okazać jedynie świadomość, że „1” to prawda, „0” zaś fałsz. Taka elementarna wiedza pozwoli jedynie na oprogramowanie systemu wbudowanego sterującego miganiem diody. Chcąc nauczać młode pokolenie logicznego myślenia, pokazać/uświadamić możliwości wykorzystania wiedzy w praktyce i jej użytecznego charakteru, musimy spojrzeć na problem nauczania podstaw, do jakich należy algebra Boole’a. Szerzej – to podstawa podstaw, od której wszystko się zaczyna. W dobie postępującej cyfryzacji ma to szczególne znaczenie.

## Podsumowanie

Brak nam na razie wiedzy na temat, jaki model kształcenia informatycznego zostanie przyjęty w podstawie programowej dla szkół ponadpodstawowych. Jeśli ambicją rządzących jest wyposażenie uczniów w rzetelne i przydatne (również

---

<sup>5</sup> Bramki logiczne nazywane są często także funktorami logicznymi.

na rynku pracy) kompetencje, powinno się odejść od dotychczasowych rozwiązań na rzecz rozwijania praktycznych umiejętności programistycznych dla potrzeb nie tylko typowo informatycznych (różnorakie aplikacje), ale przede wszystkim przydatnych w elektronice, robotyce i automatyce.

## Literatura

<http://sjp.pl/logika> (4.05.2017).

Kant, I. (2009). *Krytyka czystego rozumu*. Warszawa: Hachette.

Kaula, R. (2011). *Podstawy automatów cyfrowych*. Gliwice: Wyd. PŚ.

Kofler, E. (1956). *Z dziejów matematyki*. Warszawa: Wiedza Powszechna.

Marciszewski, W. (1988). *Mała encyklopedia logiki*. Wrocław, Warszawa, Kraków, Gdańsk, Łódź: Zakład Narodowy im. Ossolińskich.

Olszewski, A. (2017). *Wykłady z logiki dla roku pierwszego*. Pobrane z: <http://www.obi.opoka.org.pl/olszewski/pdf/LogikaIrok.pdf> (4.05.2017).

Piecuch, A. (2008). *Edukacja informatyczna na początku trzeciego tysiąclecia*. Rzeszów: Fosze.

Piecuch, A. (2016a). Nowe wyzwania przed edukacją informatyczną w świetle planowanej reformy systemu edukacji 2017/18. *Technika a vzdelávanie*, 2, 33–36.

Piecuch, A. (2016b). Programowanie dla najmłodszych. *Edukacja – Technika – Informatyka*, 4 (18), 332–338.

Piecuch, A. (2017). Programowanie może być interesujące – platforma ARDUINO. *Dydaktyka Informatyki*, 12, 155–160.

Rozporządzenie Ministra Edukacji Narodowej z 14.02.2017 w sprawie podstawy programowej wychowania przedszkolnego oraz podstawy programowej kształcenia ogólnego dla szkoły podstawowej. Dz.U. poz. 356.

Tkaczyk, M. (2017). *Logika. Wykład kursoryczny*. KUL. Pobrane z: <https://www.kul.pl/files/233/wyklady/wyklad.pdf> (4.05.2017).