W książce rozważa się także klasyfikację krzywych i powierzchni stopnia drugiego.

Rozdział XII, w którym rozważa się macierze, wyznaczniki i równania liniowe, jest luźno związany z całością. Mógłby być umieszczony jako rozdział ostatni, bowiem już w rozdziale XI korzysta się z macierzy i wyznaczników. Sposób przedstawiania materiału jest odpowiedni. Autor słusznie ogranicza do prostych przypadków uzasadnienia niektórych własności wyznaczników i macierzy. Zastosowanie macierzy do rozwiązań układów równań podano zbyt ogólnie. Brak przykładów sprawi trudność czytelnikowi zapoznającemu się po raz pierwszy z tym zagadnieniem.

Materiał przedstawiony w omawianej książce jest na ogół ujęty w postaci twierdzeń, które się dowodzi. Jest w tym duża przesada, bowiem wszystkie niemal wzory występujące w geometrii analitycznej — to twierdzenia. Np. na str. 55 znajdujemy trzy twierdzenia dotyczące kolejno kąta między wektorem a osią, dwiema osiami i dwoma wektorami. W § 48 jest siedem twierdzeń odnośnie do równania prostej na płaszczyźnie. Autor widocznie miał na celu zwrócenie uwagi czytelnika na założenia. Niestety, nie wszędzie podano warunki wystarczające, bo np. równanie (2) na str. 73 ma miejsce o ile $x_1 \neq x_0$, $y_1 \neq y_0$, a to założenie autor pomija. Wskutek tego pomija się ważne szczególne przypadki położenia prostej.

Po każdym rozdziale następują „pytania", które są zaletą książki, mają one szczególne znaczenie dla czytelnika studiującego zaocznie, oraz „zadania", po których następują „odpowiedzi".

Język książki jest poprawny, a opracowanie pod względem typograficznym jest staranne i przejrzyste.

Niewątpliwie omawiana tu część druga „Zarysu" będzie z pożytkiem służyć wielu czytelnikom, zwłaszcza studentom o zainteresowaniach technicznych jako pomoc w nauce matematyki wyższej.

Zygmunt Butlewski

Zdzisław Pawlak, *Automatyczne dowodzenie twierdzeń*, Biblioteka Matematyczna, t. 19, Państwowe Zakłady Wydawnictw Szkolnych, Warszawa 1965, str. 147.

Książeczka w popularny sposób zaznajamia czytelnika z możliwościami zastosowania maszyn cyfrowych do dowodzenia twierdzeń matematycznych. Składa się z dwóch części: pierwsza podaje ogólne ujęcie tematu, w drugiej jest opisany przypadek szczególny: dowodzenie twierdzeń rachunku zdań w systemie Gentzena (w ujęciu Hao Wanga).

Jeżeli przyjmiemy, że teoria jest zbiorem jej twierdzeń (aksjomatów lub wniosków z aksjomatów), a każde twierdzenie jest napisem w określonym alfabecie, to w tym ujęciu teoria będzie po prostu pewnym podzbiorem półgrupy wolnej, generowanej przez ten alfabet. Mając zbiór aksjomatów i reguł wnioskowania, można zupełnie automatycznie wyznaczać elementy tego podzbioru lub sprawdzać, czy jakiś element półgrupy do tego zbioru należy. Każdemu wywodowi można przyporządkować drzewo, w którym każdemu wierzchołkowi odpowiada reguła wnioskowania, łuki wchodzące do niego przesłankom, a łuk wychodzący wnioskowi tej reguły. Drzewo to można interpretować jako opis procesu sekwencyjnego; pojęcie to zostało wprowadzone już w innej książce Autora[1]. Dalej jest podany dość szczegółowy opis maszyn

[1] Z. Pawlak, *Organizacja maszyn bezadresowych*, PWN, Warszawa 1965. Zob. też recenzję w Wiadomościach Matematycznych, IX. 2, str. 286-288.

bezadresowych i uproszczonej maszyny cyfrowej adresowej, ich programów działania przy dowodzeniu twierdzeń.

Jak się wydaje, celem Autora było przede wszystkim pokazanie, że automatyczne dowodzenie twierdzeń jest możliwe. Cel ten oczywiście został osiągnięty, i to za pomocą prostych i łatwych środków. Trudno się jednak oprzeć wrażeniu pewnej jednostronności ujęcia: wszystkie przedstawione w książce programy są napisane w językach wprowadzonych tam maszyn i mniej obznajmiony z problematyką maszyn cyfrowych czytelnik mógłby odnieść wrażenie, że to właśnie jest najwłaściwszy lub najczęstszy sposób programowania maszyn cyfrowych, tym bardziej, że dość obszerne komentarze, przytoczone w książce, nie wyprowadzają go z tego błędu. W rzeczywistości korzystanie z języka maszyny jest najmniej dogodnym i najmniej przejrzystym sposobem pisania programów, stosowanym tylko w ostateczności — gdy okoliczności zmuszają do korzystania ze źle oprogramowanej maszyny, nie mającej odpowiednich języków programowych. Dzięki zastosowaniu języków algorytmicznych w programowaniu, budowa maszyny (opisywana z dokładnością do pojedynczych rejestrów) staje się czynnikiem nieistotnym lub w najgorszym razie drugorzędnym — najistotniejsza jest budowa samego algorytmu i jego realizacja w języku programowym, który z rzeczywistą budową maszyny ma na ogół bardzo mało wspólnego. Sądzę, że parę odpowiednich zdań komentarza przywróciłoby przedstawianemu obrazowi właściwe proporcje, umożliwiając czytelnikowi wyrobienie sobie właściwego poglądu w tej sprawie. Inną jest rzeczą, że metody, zastosowane przez Autora w programowaniu w języku maszyny, mogłyby być z powodzeniem stosowane w programach pisanych w językach algorytmicznych; ale o tym również nie ma w książce wzmianki. W świetle tych uzupełnień innego znaczenia nabrałaby uwaga na str. 139, że dla zaprogramowania najprostszego wnioskowania trzeba napisać setki symboli. Racja, ale rzeczywistość mimo wszystko nie jest tak przerażająca, jak to mógłby sobie wyobrazić czytelnik (niespecjalista), patrząc na program napisany na dwu poprzednich stronach.

Zupełnie nie mogę się zgodzić z uwagą Autora, podaną w odsyłaczu na str. 95, że „... zamiast pisać bezpośrednio programy realizujące zadane reguły wnioskowania, wygodniej jest podać najpierw realizację reguł wnioskowania w maszynie Turinga, a potem dopiero ułożyć programy dla maszyny cyfrowej naśladujące działanie maszyny Turinga." Maszyny Turinga są wyjątkowo niewdzięcznym narzędziem, jeżeli chodzi o wykonanie na nich jakichkolwiek praktycznych obliczeń i symulowanie ich na maszynie cyfrowej poważnie utrudniłoby napisanie programów i wielokrotnie przedłużyłoby czas liczenia, nie dając w zamian żadnych istotnych korzyści.

Stanisław Waligórski

Zdzisław Pawlak, *Gramatyka i matematyka*, Biblioteka Matematyczna, t. 20, Państwowe Zakłady Wydawnictw Szkolnych, Warszawa 1965, str. 112.

Mając dowolny zbiór skończony, można utworzyć generowaną przez niego półgrupę wolną. Podzbiory takiej półgrupy można opisywać w rozmaity sposób — w szczególności mogą do tego służyć różnego rodzaju gramatyki. Są one zwykle tworzone tak, że stanowią jednocześnie opis algorytmu konstruowania elementów tego podzbioru (gramatyki syntetyczne), lub sprawdzania czy wybrany element półgrupy do tego zbioru należy (gramatyki analityczne). Przy badaniu języków naturalnych przyjmuje się jako zbiór generatorów wolnej półgrupy cały słownik języka. W przypadku języków sztucznych generatorami są odpowiednie symbole lub napisy.