

TOMASZ NOWAK

ALGORYTMIZACJA GRAMATYKI I ORTOGRAFII – TEORIA ORAZ JEJ WYKORZYSTANIE W PRAKTYCE

WSTĘP

Niniejsza praca stanowi w zamyśle autora skromny przyczynek do propagowania osiągnięć współczesnych nauk formalnych na terenie językoznawstwa oraz dydaktyki języka polskiego. Przedstawione rozwiązania nie pretendują do rangi myśli odkrywczych i zupełnie oryginalnych. Są one raczej próbą przybliżenia problematyki algorytmu oraz możliwości jej zastosowania w procesie nauczania wybranych działów gramatyki i ortografii. Wkład odautorski sprowadza się do gruntownego zreferowania literatury przedmiotu, a także zrealizowania wybranych postulatów logiki matematycznej i lingwistyki na gruncie dydaktyki języka.

1. ALGORYTMY W LOGICE I MATEMATYCE

1.1. Historia algorytmu

Algorytm jest jednym z podstawowych pojęć współczesnych nauk formalnych (logika, matematyka). Posługiwano się już nim intuicyjnie we wcześniejszych stadiach rozwoju matematyki (starożytny Egipt, Grecja, Persja). Próbowano wówczas tworzyć procesy obliczeniowe o charakterze czysto mechanicznym. Szukane wielkości w zadaniach obliczano zatem krok po kroku na podstawie wielkości wyjściowych i zgodnie z określonymi regułami (Ałfierowa, 1977).

Mgr TOMASZ NOWAK – doktorant Uniwersytetu Śląskiego na Wydziale Filologicznym, Instytut Języka Polskiego, Pracownia Leksykografii i Metaleksykografii, e-mail: algorytm@interia.pl

Termin „algorytm” (łac. *algorithmus*) wywodzi się z połączenia greckiego wyrazu *arithmos* (liczba) i arabskiego słowa *algorism*, które oznaczało średniowieczną sztukę rachowania na liczbach zapisanych cyframi alfabetu arabskiego w systemie dziesiętnym. Pochodzenie słowa *algorism* wiąże się powszechnie z nazwiskiem średniowiecznego matematyka uzbeckiego Muhameda ibu-Musy al-Chorezmi, który w IX wieku (824 r.) opisał zasady wykonywania takiego rachunku (Marciszewski, 1987).

Paradygmatem metod efektywnych są algorytmy matematyczne. Znaczenie algorytmu najczęściej wiązano z algorytmami Euklidesa (Heller, 1980). Przykładem matematycznego algorytmu Euklidesa jest procedura znajdowania największego wspólnego dzielnika dwóch liczb dodatnich, które nie są relatywnie pierwsze (to znaczy takich, które mają wspólny dzielnik różny od 1).

Do początku trzydziestych lat ubiegłego wieku nie istniała żadna ogólna teoria algorytmów. Sytuacja uległa zmianie, kiedy na plan pierwszy wysunęły się zagadnienia algorytmiczne, których pozytywne rozwiązanie było wątpliwe. Okazało się bowiem, że czym innym jest udowodnienie istnienia algorytmu, a czym innym jego brak. Dostarczenie definicji algorytmu stało się więc jednym z podstawowych problemów matematycznych. Zaproponowano wówczas trzy sposoby rozwiązania zaistniałej aporii (Blikle, 1971):

- rozwiązanie oparte na pojęciu funkcji rekurencyjnych, czyli pewnej szczególnej klasy funkcji arytmetycznych (K. Gödel, A. Church),
- rozwiązanie oparte na opisie ściśle wyznaczonej klasy procesów (A. Turing, E. Post),
- rozwiązanie oparte na określeniu algorytmu jako szczególnej odpowiedniości między słowami w pewnym alfabecie abstrakcyjnym (A. A. Markow).

Początkowo główną dziedziną zastosowań teorii algorytmów była logika matematyczna, podstawy matematyki, algebra, geometria oraz analiza matematyczna. Współcześnie teoria algorytmów jest silnie związana z wieloma działami lingwistyki, ekonomii, fizjologii mózgu i psychologii oraz nauk przyrodniczych.

1.2. Problem rozstrzygalności

W centrum moich zainteresowań sytuuje się pojęcie algorytmu, rozumiane jako ścisły odpowiednik intuicyjnego pojęcia procedury efektywnie rozstrzygalnej. Kwestia rozstrzygalności (niem. *Entscheidungsproblem*) sprowadza się do zagadnienia: czy istnieje metoda pozwalająca rozstrzygnąć, czy dane zdanie jest dowodliwe (Hodges, 1998).

Przyjmuje się, że pewna własność jest rozstrzygalna (np. własność bycia wyrażeniem spójnym syntaktycznie) wtedy i tylko wtedy, gdy klasa pytań

„Czy x -owi przysługuje własność P ?” (dla wszystkich x z danej dziedziny) jest rozstrzygalna. Natomiast klasa pytań jest rozstrzygalna wtedy i tylko wtedy, gdy istnieje efektywna procedura, która określa odpowiedź na każde pytanie tej klasy. Na przykład: klasa pytań, „Czy x jest wyrażeniem spójnym syntaktycznie?” (gdzie x jest dowolnym wyrażeniem języka, utworzonym zgodnie z regułami derywacji gramatyki kategoryjnej), jest rozstrzygalna. Dla danego wyrażenia x można bowiem sprawdzić (za pomocą efektywnej procedury skreślania), czy jest ono redukowalne do pojedynczego symbolu kategorii (prostego lub ułamkowego), czy nie (Ajdukiewicz, 1985).

Efektywna procedura rozstrzygania (mechaniczna, algorytmiczna, automatyczna) powinna być zadana przez skończony zbiór prostych reguł, które określają ciąg elementarnych czynności, wykonywanych krok po kroku w taki sposób, że każda zależy tylko od wyniku poprzednich, a zależność ta jest całkowicie opisana przez reguły i nie wymaga żadnego „rozumienia” (Marciszewski 1987). Efektywna procedura rozstrzygania musi się zakończyć po skończonej liczbie kroków, dając odpowiedź na pytanie zadane przed pierwszym krokiem.

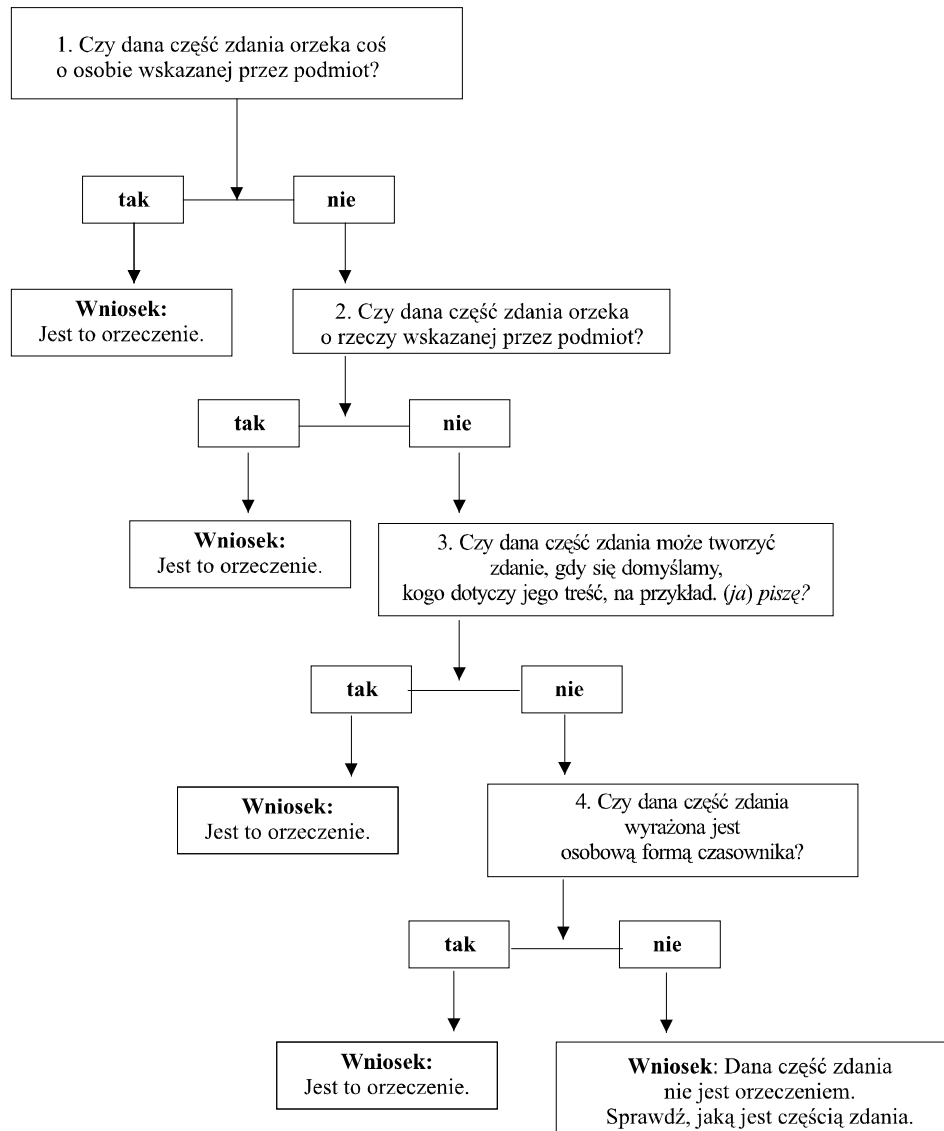
Zaproponowano różne ściśle odpowiedniki intuicyjnego pojęcia procedury efektywnej. Najwcześniej wprowadzono pojęcie funkcji rekurencyjnej, maszyny Turinga oraz algorytmu normalnego Markowa (Blikle, 1971). Okazało się także, że wszystkie te systemy formalne są równoważne w tym sensie, że klasa wszystkich funkcji definiowanych w każdym z tych systemów pokrywa się z klasą funkcji rekurencyjnych, por. tezę Churcha: „każda relacja intuicyjnie rozstrzygalna jest rekurencyjna”.

Rekapitulując: efektywna metoda rozwiązywania danego problemu to taka metoda uzyskiwania (obliczania) odpowiedzi, która – jeżeli jest stosowana poprawnie i dostatecznie długo – w sposób logicznie konieczny prowadzi do prawdziwej odpowiedzi w skończonej liczbie kroków. Warunkiem koniecznym stosowania efektywnej metody rozwiązywania jest to, że musi być ona wykonalna w sposób mechaniczny (tj. bez odwoływania się do wyobraźni czy pomysłowości użytkownika).

1.3. Praktyczne wykorzystanie rezultatów matematycznej teorii algorytmów

W dalszej części pracy będę się starał wykazać przydatność algorytmu, jako narzędzia matematycznego, w takich dziedzinach wiedzy, jak lingwistyka oraz dydaktyka języka. Teraz jednak chciałbym odnieść poczynione dotąd spostrzeżenia na temat algorytmu do zaproponowanej przeze mnie metody algorytmizacji wybranych działów gramatyki i ortografii.

Jeżeli chcesz znaleźć orzeczenie w zdaniu, to sprawdź:

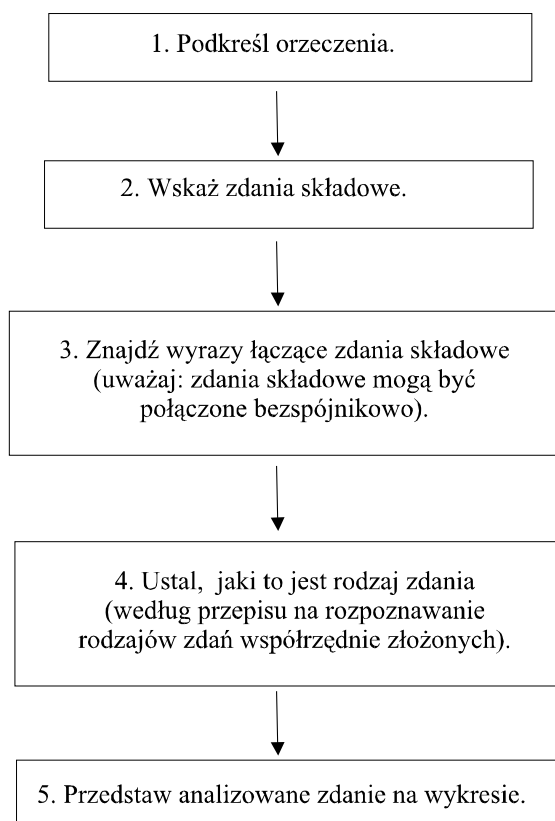


Algorytmy, które proponuję w tej pracy jako metodę dydaktyczną, są procedurami efektywnie rozstrzygalnymi, o czym można się przekonać, analizując budowę algorytmu rozpoznającego w zdaniu orzeczenie. Własność bycia orzeczeniem jest rozstrzygalna, ponieważ klasa pytań „Czy x -owi przysługuje własność bycia orzeczeniem?” jest rozstrzygalna. Pytania 1-4, wchodzące w obręb tej klasy, obejmują jedynie te własności, które charakteryzują orze-

czenie. Każde z tych pytań jest rozstrzygalne, ponieważ odpowiadając na nie za każdym razem łatwo wskazać odpowiedź. Biorąc pod uwagę to wszystko, co dotąd zostało powiedziane, stwierdzam, że skonstruowany algorytm stanowi procedurę efektywnie rozstrzygalną (obliczalną).

Zaproponowane przeze mnie algorytmy są zadane przez skończony zbiór prostych reguł, które określają ciąg elementarnych czynności, jakie należy wykonać, aby rozwiązać postawiony problem. Chcąc się o tym przekonać, wystarczy przyjrzeć się budowie algorytmu, który modeluje rozbiór zdań współrzędnych. Pięć nieskomplikowanych reguł wyznacza tutaj kolejność przeprowadzania pięciu elementarnych czynności.

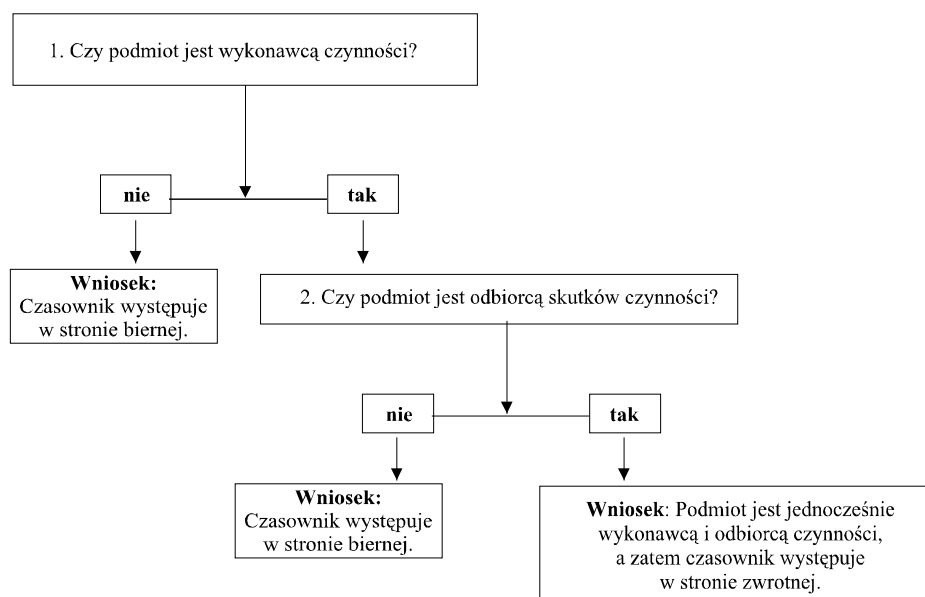
Jeżeli chcesz przeprowadzić rozbiór zdania współrzędnego, to:



Elementarne czynności, opisane przez reguły algorytmu, są wykonywane krok po kroku w taki sposób, że każda z nich zależy od wyniku poprzednich

(por. algorytm na rozpoznawanie strony czasownika, gdzie wykonanie czynności 2. jest uzależnione od rezultatu czynności 1).

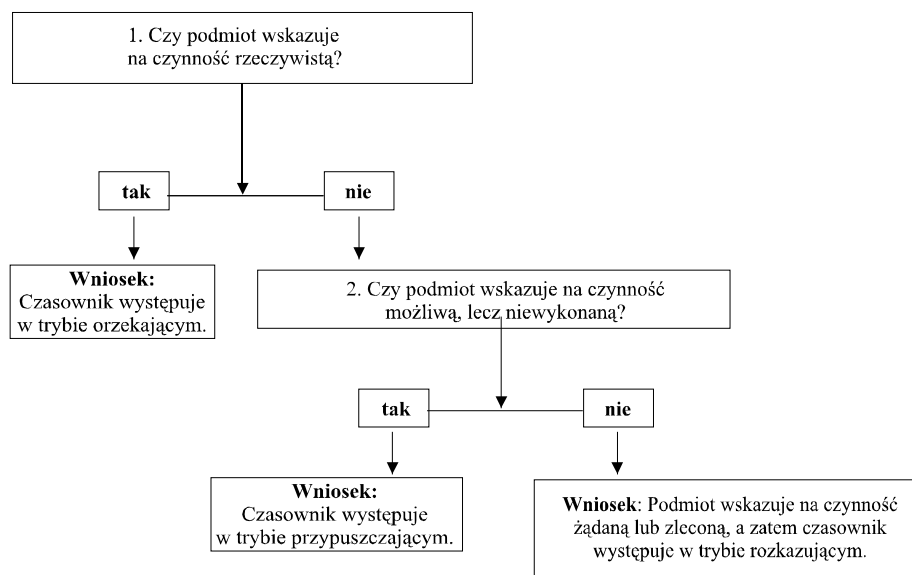
Jeżeli chcesz rozpoznać, w jakiej stronie występuje dany czasownik, to sprawdź:



Algorytm, stanowiąc efektywną procedurę rozstrzygnięcia, kończy swoją pracę po skończonej liczbie kroków, dając odpowiedź na pytanie zadane przed pierwszym krokiem. Algorytmiczna metoda rozwiązywania problemów gramatycznych (uzyskiwania odpowiedzi)¹ stosowana poprawnie i dostatecznie długo, w sposób logicznie konieczny prowadzi zawsze do prawdziwej odpowiedzi w skończonej liczbie kroków. Własność tę ilustruje algorytm rozpoznający tryb czasownika, który po dwóch krokach udziela odpowiedzi na pytanie o tryb, w jakim występuje dany czasownik.

Jeżeli chcesz rozpoznać, w jakim trybie występuje dany czasownik, to sprawdź:

¹ Procedurę odpowiadania na poszczególne pytania można porównać do funkcji matematycznej o argumentach należących do dziedziny poprawnie zbudowanych wyrażeń języka polskiego oraz wartościach „tak” i „nie”.



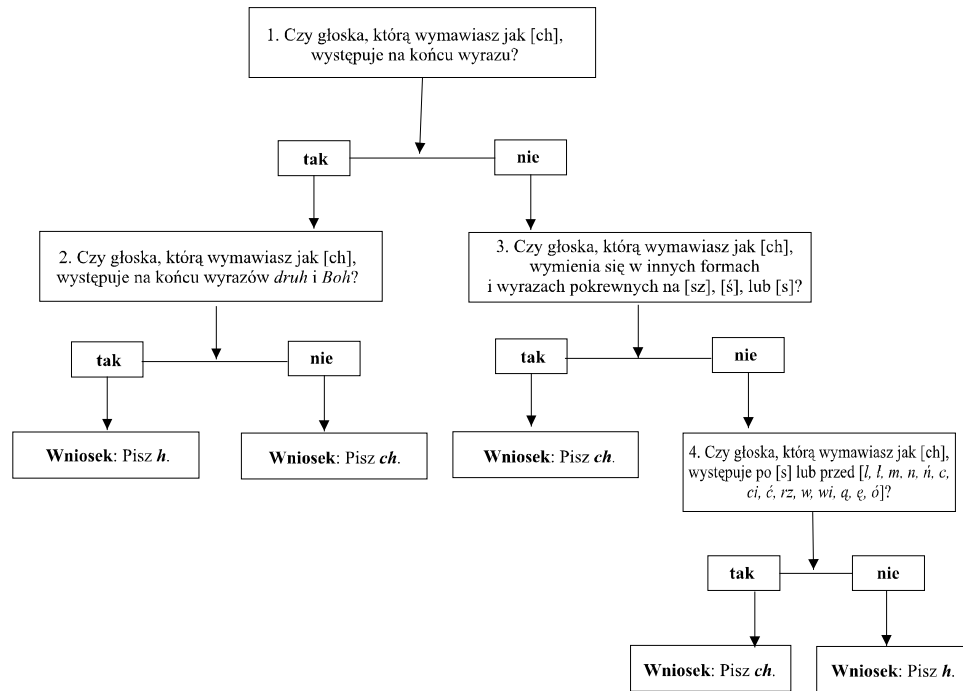
Proces logicznej dedukcji, z jakim mamy do czynienia w każdym algorytmie, zostaje zredukowany do ciągu bardzo prostych i dających się mechanicznie wykonywać przekształceń, określonych na ciągach symboli w pewnym skończonym alfabecie.

Poniżej prezentuję własny algorytm pisowni wyrazów z „ch” i „h”, przedstawiony w postaci sieci działań oraz sformalizowany (na podstawie prac L. N. Łandy i J. Porayskiego-Pomsty) w języku konstruktywnej matematyki rekurencyjnej².

ALGORYTM PISOWNI WYRAZÓW Z „h” i „ch”

Jeżeli chcesz ustalić, czy w danym wyrazie napisać literę „ch” czy „h”, to sprawdź:

² Przedmiot konstruktywnej matematyki rekurencyjnej stanowią obiekty konstruktywne, czyli słowa budowane ze znaków jakiegoś wyjściowego alfabetu. Obiekty konstruktywne powstają przez ciągi kolejnych wyborów, których dokonuje się na podstawie reguł algorytmu. Konstruktywna matematyka rekurencyjna konsekwentnie zakłada stanowisko nominalistyczne i reprezentuje orientację językową (Murawski, 1995). Zapewne dlatego jej osiągnięcia są wykorzystywane we współczesnej lingwistyce formalnej.



ALGORYTM PISOWNI WYRAZÓW Z „h” i „ch”

1. ALFABET ABSTRAKCYJNY:

Alfabet nieterminalny (kolejne pytania algorytmu): $V_n = \{A, B, C, D\}$

Alfabet terminalny (kolejne wnioski algorytmu): $V_t = \{a, b, c, d, e\}$

Alfabet wejściowy: $V_i = \{A, B, C, D, a, b, c, d, e\}$

Alfabet wyjściowy: $V_e = \{a, b, c, d, e\}$

Słowo wejściowe: zdanie, które należy do klasy zdań pojedynczych:

$ABabCDcd$

Operator alfabetowy wieloznaczny: $\Gamma(ABabCDcd)$

$\Gamma(ABabCDcd) = abababcd$

$\Gamma(ABabCDcd) = ababbacd$

$\Gamma(ABabCDcd) = cdabcdcd$

$$\Gamma(ABabCDcd) = acdaabcdcd$$

$$\Gamma(ABabCDcd) = adcaabcdcd$$

Reguły:

$$AB \rightarrow aBa \quad AB \rightarrow cd$$

$$AB \rightarrow dc \quad B \rightarrow cd$$

$$B \rightarrow dc \quad CD \rightarrow ab$$

$$CD \rightarrow ba \quad CD \rightarrow cd$$

2. IDENTYFIKATORY I OPERATORY

Przyjmuję następującą konwencję: identyfikatory otrzymują indeksy liczbowe, rosnące w miarę realizacji algorytmu – parzyste (identyfikator „nie”) i nieparzyste (identyfikator „tak”). Odpowiednio są oznaczone operatory, wywoływane przez przyporządkowane im identyfikatory.

$$IE_1 = \text{tak}, \quad OE_1 = AB \rightarrow ab, \quad ABabCDcd \rightarrow ababCDcd,$$

$$IE_3 = \text{tak}, \quad OE_3 = CD \rightarrow cd, \quad ababCDcd \rightarrow abababcd,$$

$$IE_4 = \text{nie}, \quad OE_4 = CD \rightarrow ba, \quad ababCDcd \rightarrow ababbacd,$$

$$IE_2 = \text{nie}, \quad OE_2 = CD \rightarrow cd, \quad ABabCDcd \rightarrow ABabcdcd,$$

$$IE_5 = \text{tak}, \quad OE_5 = AB \rightarrow cd, \quad ABabcdcd \rightarrow cdabcdcd,$$

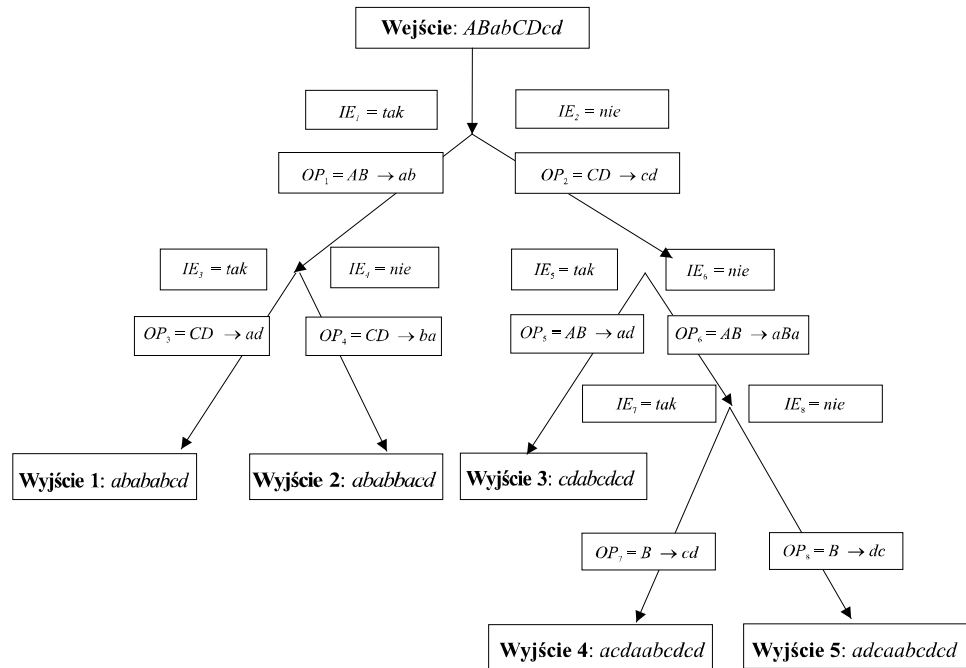
$$IE_6 = \text{nie}, \quad OE_6 = AB \rightarrow aBa, \quad ABabcdcd \rightarrow aBaabcdcd,$$

$$IE_7 = \text{tak}, \quad OE_7 = B \rightarrow cd, \quad aBaabcdcd \rightarrow acdaabcdcd,$$

$$IE_8 = \text{nie}, \quad OE_8 = B \rightarrow dc, \quad aBaabcdcd \rightarrow adcaabcdcd.$$

GRAF ZORIENTOWANY – SIEĆ DZIAŁAŃ

Skonstruowany przeze mnie algorytm stanowi egzemplifikację teoretycznych rozważań, jakim poświęciłem pierwszy rozdział. Jak można zauważyć, logika i matematyka dostarczają narzędzi, które umożliwiają rozwiązywanie w sposób rozstrzygalny wielu zagadnień (nie tylko z zakresu nauk formalnych). Metody algorytmiczne pozwalają na ścisłe i jednoznaczne sformułowanie badanych problemów, a także na ich implementację, dzięki czemu można automatycznie, za pomocą komputera, korzystać z przedstawionego algorytmu oraz zweryfikować poprawność zaproponowanych rozwiązań.



2. ALGORYTMY W LINGWISTYCE

2.1. Lingwistyka strukturalna jako nauka o modelach lingwistycznych

Główny cel lingwistyki strukturalnej sprowadza się do wyjaśnienia różnych umiejętności językowych użytkowników języka (Apresjan, 1971), m.in.:

- umiejętności odróżniania w języku tego, co jest poprawne od tego, co nie jest poprawne;
- umiejętności wydobycia z danego zdania zawartej w nim informacji (rozumienie, analiza);
- umiejętność budowania zdania na podstawie zadanej informacji (mówienie, synteza).

Językowe umiejętności użytkowników języka nie są jednak bezpośrednio dostępne obserwacji uczonego. Stąd bierze się konieczność ich modelowania³. Centralnym pojęciem współczesnej lingwistyki strukturalnej stało się

³ Konieczność modelowania powstaje we wszystkich tych dziedzinach nauki, w których obiekt jest niedostępny bezpośredniej obserwacji. Pojęcie modelu można porównać do czarnej

zatem pojęcie modelu. Lingwistykę strukturalną określa się w związku z tym jako naukę o modelach języka.

Celem nauki jest opisywanie, wyjaśnianie i przewidywanie zjawisk, które zachodzą w otaczającym nas świecie. Rzeczywistość (także i ta językowa) jest jednak zbyt złożona, aby móc ją dokładnie opisać bez uwzględniania uproszczeń i przybliżeń. Chcąc więc opisać wybrane zjawisko, bierze się pod uwagę jedynie te elementy rzeczywistości, które mają na to zjawisko istotny wpływ. Najpierw próbuje się więc zrozumieć zachowanie bardzo prostych obiektów, licząc na to, że w ten sposób można odkryć istotne własności rządzące zachowaniem obiektów bardziej złożonych. W związku z tym uwzględnia się wszystkie te czynniki, które mają wpływ na przebieg badanych zjawisk, a następnie formułuje się warunki konieczne i wystarczające (Białynicka-Birula, Białynicki-Birula, 2002).

Modelowanie polega więc na zbudowaniu obrazu obiektu na podstawie zestawienia danych wejściowych i końcowych. Dalszy etap stanowi wysunięcie hipotezy o sposobie funkcjonowania badanego obiektu, a następnie zrealizowanie tej hipotezy w postaci algorytmu. Ostatecznie sens modelowania sprowadza się do tego, że zamiast ukrytych przed uczonym właściwości obiektu należy zbadać podane w jawnej postaci własności modelu, a następnie rozszerzyć na badany obiekt wszystkie te prawa, które zostały wyprowadzone z modelu.

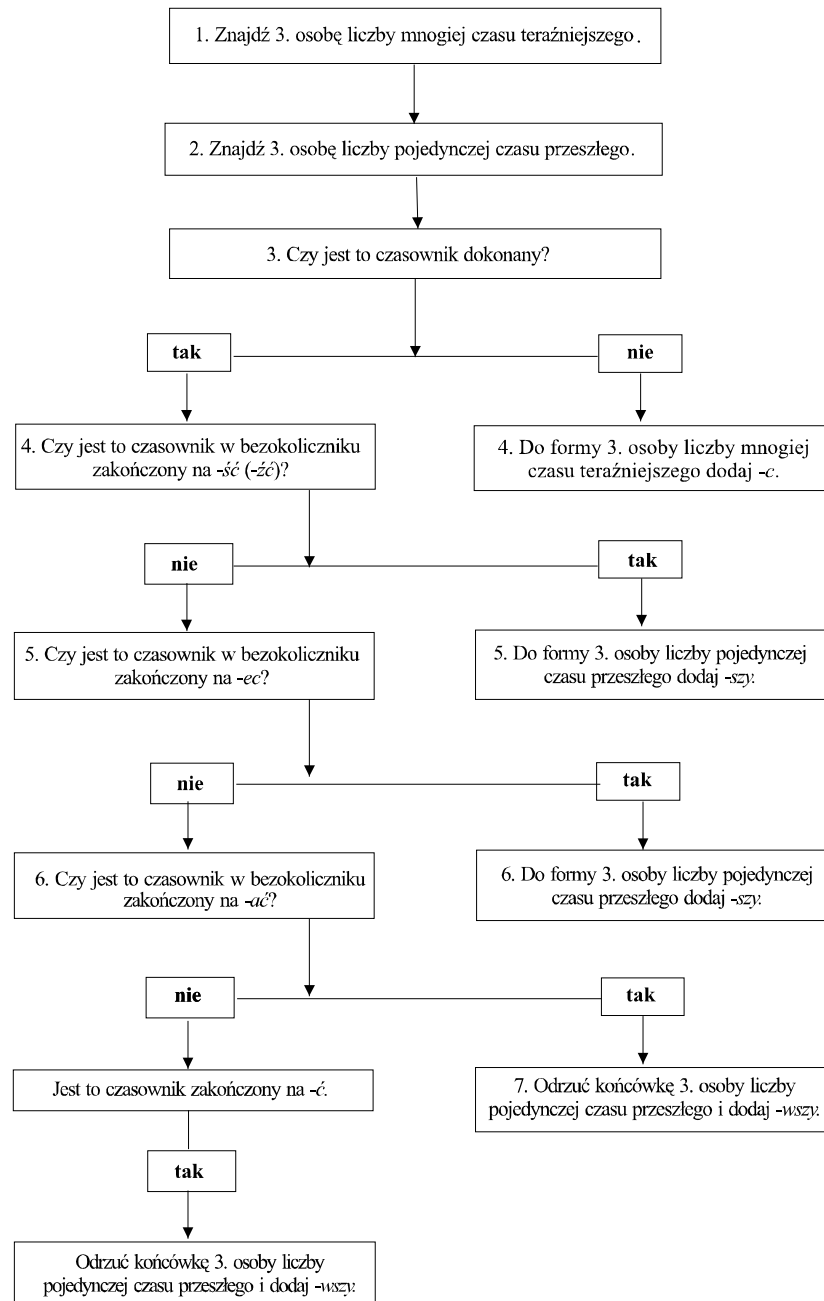
Modelowanie w lingwistyce wymaga przyjęcia założenia, że język (*langue*) jest mechanizmem, który według ściśle określonej metody przetwarza zespoły jednych jednostek w zespoły innych jednostek. Języka (*langue*) nie można jednak opisać, ponieważ nie jest on dostępny bezpośredniej obserwacji. Język (*langue*) można jedynie modelować, śledząc jego funkcjonowanie na płaszczyźnie *parole* (Porayski-Pomsta, 1981).

Modele lingwistyczne, realizowane pod postacią algorytmów, można rozmaicie klasyfikować. Podział modeli lingwistycznych można przeprowadzić w zależności od tego, który przedmiot badań stanowi obiekt danego modelu. Literatura przedmiotu wyszczególnia wobec tego następujące typy modeli lingwistycznych:

1. modele imitujące działalność językową człowieka, czyli takie modele, których przedmiot stanowią konkretne procesy i zjawiska językowe, por. algorytm tworzenia imiesłówów przysłówkowych (Słomkiewicz, 1972).

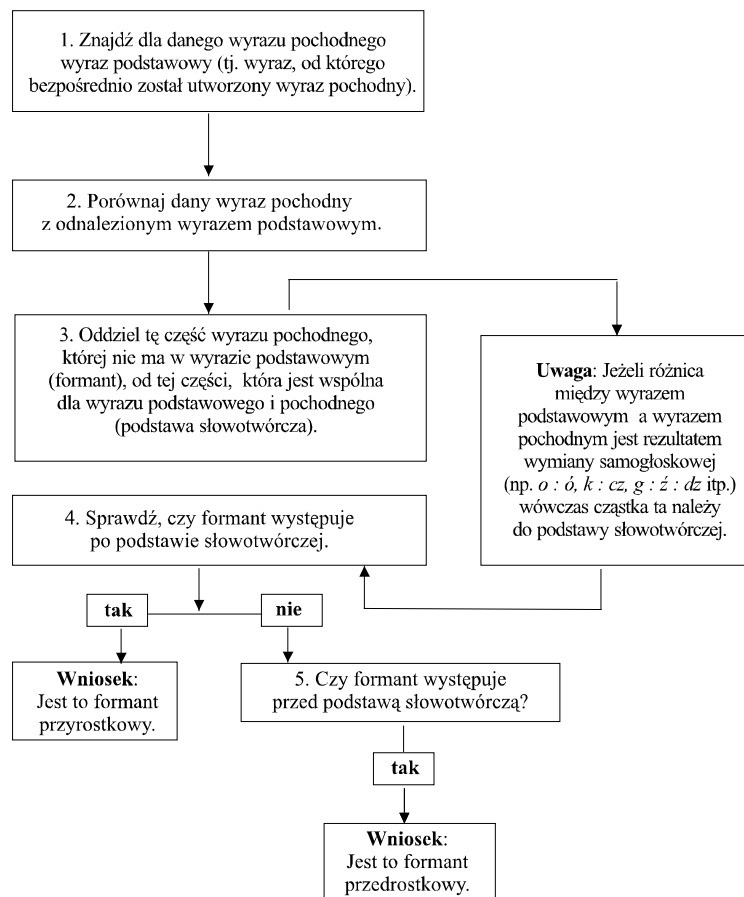
skrzynki, o której wiadomo tylko, jakie materiały początkowe otrzymuje na wejściu i jakie produkty końcowe wydaje na wyjściu. Zadanie badacza sprowadza się zatem do tego, aby dowiedzieć się, co zawiera czarna skrzynka, czyli ukryty przed badaczem mechanizm, który przerabia materiały wejściowe w produkty końcowe (Porayski-Pomsta, 1981).

Jeżeli chcesz utworzyć imiesłów przysłówkowy od danego czasownika, to:



2. modele badawcze, które imitują działalność badawczą lingwisty – modelowany obiekt stanowią tutaj procedury, które prowadzą uczonego do opisu konkretnego zjawiska językowego, por. algorytm analizy słowotwórczej wyrazów pochodnych.

Jeżeli chcesz przeprowadzić analizę słowotwórczą danego wyrazu pochodnego, to:



3. modele meta-teoretyczne, w których przedmiotem modelowania są gotowe opisy lingwistyczne.

Oczywiście, najważniejszym typem modeli lingwistycznych są modele, które imitują działalność językową człowieka (w stosunku do tych ostatnich modele badawcze i meta-teoretyczne pełnią jedynie funkcję pomocniczą).

Klasyfikację modeli, które imitują działalność językową, można z kolei przeprowadzić w zależności od tego, która strona działalności językowej (rozumienie bądź mówienie) jest przedmiotem modelowania (Grzegorzczkowska, 2001). W związku z tym wyróżnia się:

1. modele syntezy, realizowane w postaci algorytmów generacji, które obejmują skończoną liczbę reguł, nadających się do skonstruowania nieskończenie wielu poprawnych zdań,
2. modele analizy, realizowane w postaci algorytmów analizy, które obejmują skończoną liczbę reguł, pozwalających na zanalizowanie nieskończenie wielu zdań danego języka.

2.2. Algorytmy generacji i analizy języków

Przedmiot moich rozważań stanowi w tej pracy pojęcie algorytmu oraz sposób jego wykorzystania na obszarze dydaktyki i lingwistyki. Algorytm lingwistyczny to taki ciąg poleceń, których spełnienie prowadzi do zbudowania bądź wyróżnienia żadanego obiektu językowego⁴. Literatura przedmiotu wyszczególnia dwa rodzaje algorytmów lingwistycznych (Semiński-Polkowska, Polkowski, 1994):

⁴ Algorytmy są konstruowane drogą wnioskowania dedukcyjnego opartego na regule *modus ponendo ponens*: jeżeli $A \rightarrow B$ jest prawdziwe i A jest prawdziwe, to B jest prawdziwe. Istnieją dwa sposoby korzystania z tej reguły w trakcie konstrukcji algorytmów – dedukcja progresywna oraz dedukcja regresywna. W obu przypadkach wiemy, że poprzednik implikacji jest prawdziwy i sprawdzamy poprawność następnika implikacji.

W przypadku dedukcji progresywnej wychodzimy od poprzedników i próbujemy wywnioskować (wyprowadzić, wygenerować) następniki. Taki sposób postępowania jest w istocie rodzajem obliczania symbolicznego (tj. manipulacji zapisami symbolicznymi, o których wiemy, że są prawdziwe, tak aby wygenerować, zgodnie z regułą *modus ponendo ponens*, nowe zapisy).

Natomiast w przypadku dedukcji regresywnej najpierw formułujemy konkluzję, a dopiero potem próbujemy ją uzasadnić poprzez znalezienie w modelu prawdziwych zapisów, z których za pomocą reguły *modus ponendo ponens* można wywnioskować naszą konkluzję (Borkowski, 1991).

Kierunek dowodzenia twierzeń w algorytmach generacji pokrywa się z kierunkiem wnioskowania – mamy tu bowiem do czynienia z rozumowaniem dedukcyjnym progresywnym i odkrywczym. Wnioskowanie jest rozumowaniem progresywnym, tj. takim, w którym dana jest racja (typ), a szuka się następstwa (wyrażenie). Wnioskowanie jest także rozumowaniem odkrywczym, tj. takim, że dane jest zdanie już uznane (typ), a poszukuje się zdania dotychczas nieuznanego (wyrażenie).

Kierunek sprawdzania twierzeń (akceptowania wyrażań) w algorytmach analizy językowej pokrywa się z kierunkiem dowodzenia (rozumowanie dedukcyjne regresywne, uzasadniające). Dowodzenie jest rozumowaniem regresywnym, tj. takim, w którym dane jest następstwo (wyrażenie), a szuka się racji (typ). Dowodzenie jest także rozumowaniem uzasadniającym, tj. takim, że dane jest zdanie jeszcze nie uznane (wyrażenie), a poszukuje się zdania już uznanego (typ).

1. algorytmy generacji języków, czyli gramatyki formalne, które działają w trybie generującym, wyprowadzając wyrażenia z ustalonego typu początkowego, por. gramatyki transformacyjno-generatywne, gramatyki struktur frazowych (HPSG, GPSG), gramatyki logiczne;

2. algorytmy analizy języków, czyli automaty, które działają w trybie rozpoznającym, przechodząc od wyrażenia (na wejściu) do typu (na wyjściu), por. gramatyki kategoryjne, modele formalno-dystrybucyjne.

Algorytmy generacji języków, czyli gramatyki formalne, to narzędzia opisu budowy języka za pomocą skończonego zbioru reguł konstrukcji i transformacji. Natomiast automaty to algorytmy analizy syntaktycznej, które otrzymują dowolne wyrażenie jako wejście i w skończonej liczbie kroków produkują wyjście, będące opisem strukturalnym tego wyrażenia lub tylko prostą odpowiedzią „tak”, gdy badane wyrażenie jest poprawne, lub „nie”, gdy badane wyrażenie nie jest poprawne (Buszkowski, 1999)⁵. Rzecz jasna, każdy algorytm może być stosowany zarówno w trybie generacji, jak i analizy (formalnie, wystarczy odwrócić kierunek strzałek w regułach)⁶.

2.3. Algorytm jako model kompetencji językowej

W związku ze stosowaniem w lingwistyce i dydaktyce procedur algorytmicznych powstaje pytanie, czy algorytmy rzeczywiście modelują zachowania mowne użytkowników języka.

⁵ Człowiek, dla którego dany język jest językiem ojczystym, może dostatecznie szybko ocenić gramatyczną poprawność przedstawionej mu frazy w tym języku. Można więc założyć, że dysponuje on pewnym algorytmem rozpoznającym.

⁶ Modelowanie języków stanowi główny kierunek badań lingwistyki matematycznej – dyscypliny naukowej ściśle związanej zarówno z językoznawstwem, jak i matematyką. Lingwistyka matematyczna powstała w związku z przeświadczeniem, że u podstaw języka leżą ściśle reguły. Stanowi ona metateorię w stosunku do językoznawstwa w tym sensie, że opisuje i bada metodami matematycznymi najbardziej uniwersalne własności konstrukcji językowych. Formalne narzędzia badania i opisu, jakich używa logika i matematyka, gwarantują precyzyjny opis języka (Buszkowski, 1999).

Pojęcia i metody lingwistyki matematycznej można stosować do wszelkich możliwych języków – w tym także do języków naturalnych. Języki naturalne można bowiem rozpatrywać jako zbiory zdań, czyli zbiory skończonych ciągów utworzonych z elementów pewnego bazowego słownika. Gdy język jest skończony, można go dokładnie określić, wyliczając jego elementy. Nie jest to natomiast możliwe, gdy jest on nieskończony. Lingwistyka matematyczna zajmuje się zatem skończonymi reprezentacjami dla języków nieskończonych. Narzędziem, spełniającym podane kryteria, jest właśnie algorytm, czyli skończony zbiór reguł wyróżniających (tj. produkujących bądź rozpoznających) wszystkie ciągi nad danym słownikiem należące do pewnego języka (Buszkowski, 1999).

J. D. Apresjan porównuje algorytm do instrukcji, którą laborant otrzymuje od swojego profesora (Apresjan, 1971). Porównanie to zakłada, że laborant dokładnie wykonuje polecenia swojego przełożonego, nie robi nigdy błędów, lecz nie jest zdolny do samodzielnego myślenia. Instrukcja dla laboranta obejmuje rozkazy typu „znajdź przymiotnik”. Porównanie Apresjana wielokrotnie poddawano krytyce. Nawet najdokładniejsza instrukcja nie zwalnia bowiem laboranta od odpowiedzialności, samokontroli i autorefleksji. Maszyna wykonuje bowiem tylko konkretne operacje typu „znajdź przymiotnik”, „zbuduj zdanie złożone podrzędne przydawkowe” itp. Laborant znajduje się jednak w nieco innej sytuacji niż maszyna cyfrowa, ponieważ dla automatu rozkazy nic nie znaczą, natomiast znaczą one wiele dla laboranta.

Podobne stanowisko w sprawie algorytmizacji procesów językowych zajmuje J. R. Searle, proponując eksperyment myślowy, zwany „chińskim pokojem” (Heller, 1980). Eksperyment ten można streścić w następujący sposób. Załóżmy, że „tłumacz” jest zamknięty w pokoju. Otrzymuje on kartki zapisane w języku chińskim, którego nie zna, wraz ze sformalizowanymi instrukcjami umożliwiającymi manipulację zapisami i generację nowych zapisów w języku chińskim. Otrzymywane kartki odpowiadają opowiadaniom i pytaniom ułożonym do tych opowiadań. Sformalizowane instrukcje są odpowiednikami algorytmów (programów komputerowych), natomiast generowane za ich pomocą zapisy są odpowiedziami na te pytania. Tak więc „tłumacz” jest w stanie „udzielać odpowiedzi” na pytania dotyczące opowiadań, których nie rozumie. Załóżmy, że w drugiej fazie eksperymentu „tłumacz” otrzymuje opowiadania w języku angielskim, który rozumie, wraz z pytaniami do nich. Podobnie jak poprzednio, udziela odpowiedzi na te pytania. Przyjmijmy teraz, że po pewnym czasie „układacze instrukcji” (czyli odpowiednicy programistów komputerowych) doszli do takiej perfekcji w pisaniu instrukcji, że odpowiedzi w języku chińskim generowane za ich pomocą nie są odróżnialne od analogicznych odpowiedzi w języku angielskim. Teraz powstaje pytanie: czy mamy do czynienia z jakąś różnicą w obu przypadkach? Searle odpowiada: tak. W pierwszym przypadku „tłumacz” zachowuje się jak zaprogramowany komputer, to znaczy, nie rozumie tego, co robi (działa bowiem według określonego algorytmu). Natomiast warunkiem koniecznym, aby poznający podmiot można było określić jako myślący, jest rozumienie przez podmiot poznawanej rzeczy.

Prorocy sztucznej inteligencji (M. Minsky, E. Fredkin) twierdzili, że pracę mózgu można sprowadzić do wykonywania szeregu obliczeń według określonego algorytmu, który mógłby wykonywać komputer o odpowiednio dużej mocy (przy założeniu, że mózg działa podobnie jak komputer). Współcześnie

jednak coraz częściej taki pogląd jest podawany w wątpliwość. Badacze przychylają się raczej do zdania, że mózg ludzki działa w sposób niealgorytmiczny (tj. inaczej niż komputer). Ludzka racjonalność wykracza, ich zdaniem, poza mechaniczne stosowanie reguł wnioskowania. Na przykład: dla człowieka jest oczywiste, że suma liczb parzystych jest zawsze parzysta. Natomiast komputer działa według określonego algorytmu: weź dwie liczby parzyste, dodaj je do siebie, a następnie sprawdź, czy ich suma dzieli się przez 2; jeżeli suma dwóch liczb parzystych nie dzieli się przez 2, to zadanie zostało wykonane; natomiast jeżeli suma dwóch liczb parzystych dzieli się przez 2, to próbuj dalej. Maszyna cyfrowa sprawdza w ten sposób wszystkie możliwe pary liczb parzystych. Jej praca nigdy się nie skończy – maszyna rozwiązuje zadanie w nieskończoność.

Jak zatem widać, wyobraźnia – zarówno matematyczna, jak i przede wszystkim lingwistyczna – nie ma charakteru algorytmicznego. Nie można zatem sprowadzać matematyki (a tym bardziej lingwistyki) do zbiorów tez wyjściowych i reguł wnioskowania, za pomocą których dowodzi się twierdzeń na temat budowy języka. Gramatyki formalne, które funkcjonują w postaci systemów dedukcyjnych, nie mogą wiernie modelować kompetencji językowej, lecz jedynie ją aproksymować. Istotę języków naturalnych stanowi bowiem ich semantyka, a ta – jak wiemy – opiera się próbom formalizacji. Pierwsze twierdzenie Gödla, na którym oparto te rozważania, wnosi, że niealgorytmiczny akt rozumienia znajduje się poza zasięgiem komputera (jak i wszelkiego formalizmu). Twierdzenie o nierozstrzygalności (Krajewski, 2003) głosi, że dla każdej dostatecznie bogatej teorii, istnieje takie zdanie, iż wychodząc z założeń tej teorii nie da się wywnioskować, czy zdanie to jest prawdziwe, czy fałszywe. Jednakże pomimo niemożności dowiedzenia ludzie często czują bądź rozumieją, czy mają do czynienia z prawdą, czy fałszem. W przeciwieństwie do ludzi maszyny nie rozumieją tego, co robią. Proces rozumienia wymaga bowiem świadomości. Nie ma zatem sensu mówić „rozumieć coś”, jeżeli nie jestem tego świadomy. Rozumienie stanowi więc jeden z przejawów świadomości.

R. Penrose w swojej błyskotliwej monografii *Nowy umysł cesarza, o komputerach, umyśle i prawach fizyki* (Penrose, 1996) stawia tezę, że procesy myślowe dają się algorytmizować, ale wynik takiego algorytmu nie zawsze jest przekładalny na język, którym się porozumiewamy. Nie można równocześnie algorytmizować procesów wewnętrznych i procedury przekładu. Przedstawiony paradygmat myślenia przeciwstawia się tezie, że mózg jest swoistym procesorem cyfrowym. Jak się zatem okazuje, procedury algoryt-

miczne, stanowiąc formalną reprezentację kompetencji językowej, są skazane na nieustanną aproksymację (choćby tylko asymptotyczną).

Jakie wnioski płyną z powyższych wywodów dla tematyki mojej pracy? Dla zagadnień dydaktycznych stan badań w zakresie algorytmizacji wybranych działów gramatyki wydaje się zadowalający. Sytuacja komplikuje się, gdy przechodzimy na grunt lingwistyki teoretycznej. O ile prosty algorytm, który modeluje czynność poszukiwania orzeczenia w zdaniu, wystarcza na potrzeby edukacji ucznia, o tyle w bardziej skomplikowanych przypadkach nasze rozumienie załamuje się. Gramatyki języków naturalnych są bowiem opisami o tak wielkim stopniu złożoności, że potwierdzenie ich najlepiej byłoby powierzyć komputerom, aby te same sprawdziły, jakie zdania są opisane przez daną gramatykę i jak są opisane (Bańko, 1990)⁷. Tutaj jednak dosięga nas problem algorytmizacji procesów wewnętrznych i procedury przekładu. Aby móc zatem myśleć o holistycznym modelowaniu języków naturalnych, trzeba skonstruować taką metodologię, która posługiwałaby się wglądem niealgorytmicznym. Problematyka ta jednak przekracza łamy niniejszej rozprawy i domaga się osobnego opracowania.

⁷ Egzemplifikację pojęcia algorytmu lingwistycznego stanowią analizatory składniowe. Analizator składniowy (ang. *parser*) stanowi implementację gramatyki, czyli taki program, który umożliwia korzystanie z niej za pomocą komputera. Działanie analizatora polega na szukaniu dowodu formuły ZDANIE (x) dla danego ciągu słów, który reprezentuje analizowane zdanie. Wynikiem jest albo odrzucenie zdania, gdy dowód taki nie może być znaleziony, albo jego akceptacja. W tym drugim przypadku wynikiem analizy jest także sam dowód, zwykle przedstawiany w postaci grafu, zwanego drzewem wyvodu. Drzewo wyvodu jest zarazem opisem struktury syntaktycznej, jaką dana gramatyka przyporządkowuje analizowanemu zdaniu (Bień, 1993).

Analizatory składniowe odgrywają pewną rolę w dydaktyce języka. Na podstawie *Gramatyki języka łemkowskiego* (Fontański, Chomiak, 2000) został opracowany łemkowski moduł językowy. Zawiera on podstawowy zasób wyrazów i form gramatycznych, a także daje możliwość automatycznej korekty wpisywanego tekstu wraz z propozycją poprawnej formy w miejsce błędnej. łemkowski moduł językowy jest wykorzystywany w nauczaniu języka łemkowskiego jako języka mniejszości w szkołach podstawowych.

Analizator składniowy stanowi algorytm określania składniowej struktury tekstu. Modeluje on zachowanie się ucznia, który rozbiera zdanie na lekcji. Algorytm automatycznej analizy powinien jednak modelować zachowanie się ucznia bardzo dobrego (tj. takiego, który analizuje zdania prawidłowo, w sposób nie wywołujący protestu ze strony nauczyciela). Powinien on także ustalać związki składniowe w taki sposób, aby odpowiadały one intuicyjnym odczuciom nosicieli języka. Praktyczną wartość takiego algorytmu określa się tym, na ile dobrze modeluje on językowe zachowanie się swojego użytkownika i na ile obiektywnie został przedstawiony (Bańko, 1990).

3. ALGORYTMY W PSYCHOLOGII I DYDAKTYCE

3.1. Teoria uczenia się

Teoria uczenia się zajmuje we współczesnej psychologii szczególnie ważne miejsce. „Uczenie się” to powszechne w życiu organizmów zjawisko, określane jako zmiany w zachowaniu, które powstają w wyniku przystosowywania się jednostek do zmiennych warunków życia. Wszystkie żywe istoty stawiają hipotezy dotyczące otaczającej je rzeczywistości. W zależności od tego czy przypuszczenia te są trafne, czy błędne, przedłużają im życie. Jeżeli hipoteza postawiona przez stworzenie zajęte walką o przetrwanie jest trafna, to jest też zarazem prawdziwa (i na odwrót). Gdyby bowiem okazała się fałszywa, to krok podjęty zgodnie z sugestiami co do konstrukcji świata, płynącymi z takiej hipotezy, mógłby się okazać ostatnim krokiem w życiu tej istoty (Popper, 1992).

W tym miejscu psychologia uczenia się spotyka się z ewolucjonistyczną teorią poznania, której główna teza brzmi, że to, co dane w naszym myśleniu (kantowskie *a priori*), jest wynikiem ewolucji. „Nasze stałe formy oglądu i kategorie pasują do zewnętrznego świata z tych samych powodów, z jakich kopyto konia pasuje do stepu, a płetwy ryby do wody” (Lorenz, 1977). Badanie „aparatu obrazu świata” u człowieka opiera się na fundamentalnej zasadzie: życie polega na uczeniu się (ewolucja stanowi więc proces zdobywania wiedzy).

Ludzki aparat ratiomorficzny (niem. *Weltanschauunngapparat*) formował się przez miliony lat pod wpływem doboru naturalnego. Z tego powodu nie mógł sobie pozwolić na błąd w dziedzinach istotnych dla przeżycia. Zachodzi zatem istotna zgodność naszego widzenia świata z odzwierciedlaną rzeczywistością (w tym sensie ludzka racjonalność jako biologiczne narzędzie wykrywania prawdy o świecie jest czynnikiem sprzyjającym przetrwaniu). Odziedziczone przez nas formy oglądu przestrzeni, czasu i przyczynowości charakteryzują się wysokim prawdopodobieństwem (hipoteza robocza), lecz nie powinny rościć sobie pretensji do absolutnej pewności (Kunzmann, Burkard, Wiedmann, 1999)⁸.

Człowiek to najbardziej rozwinięta istota na Ziemi. Miarę rozwoju, zgodnie z którą wszystkie istoty możemy poklasyfikować jako bardziej lub mniej rozwinięte, stanowi ilość informacji o świecie, które dany gatunek może

⁸ Nasza „reprodukcja świata” zawodzi bowiem w obliczu bardziej skomplikowanych zależności (np. mechanika falowa, fizyka jądrowa).

zgrupować, przechowywać i wykorzystywać (Wickler, 1991). Im bardziej rozwinięty gatunek, tym więcej może zebrać danych o środowisku, w którym żyje, i tym mniej jest w swoim zachowaniu uwarunkowany. Człowiek zajmuje w przyrodzie wyróżnioną pozycję, ponieważ potrafi rozróżniać o wiele więcej sytuacji, niż zostało to zaprogramowane w jego aparacie biologiczno-instynktownym. Jeżeli nie istnieje gotowe rozwiązanie, to należy rozstrzygnąć, jak postąpić. Innymi słowy: przystosowując się (ucząc się), jesteśmy skazani na myślenie.

3.2. Struktura ludzkiego myślenia

Strukturę myślenia tworzy materiał myślenia, czyli aktualnie dostępne informacje, a także operacje, które umożliwiają przetwarzanie informacji, oraz reguły wykonywania tych operacji (Kwiatkowski, 1982).

Substancję myślenia stanowią informacje, na które składają się spostrzeżenia, wyobrażenia i pojęcia. Spostrzeżenia dostarczają informacji o przedmiotach i procesach, jakie zachodzą w otaczającym nas świecie. Spostrzeżenia stanowią przy tym budulec wyobrażeń, czyli obrazów rzeczy i zjawisk, które tworzymy w swoich umysłach⁹.

Podstawę myślenia stanowią jednak te informacje, które są zakodowane pod postacią pojęć. To właśnie myślenie pojęciowe umożliwia operowanie pojęciami w oderwaniu od konkretnych cech przedmiotów lub zdarzeń. Dzięki myśleniu pojęciowemu możemy dokonywać uogólnień i syntez. Do podstawowych operacji, które umożliwiają manipulowanie informacjami, zalicza się: analizę, syntezę, porównywanie, abstrahowanie i uogólnianie.

Trzeci i ostatni element struktury myślenia stanowią reguły, które sterują kolejnością przeprowadzania operacji myślowych. W procesie myślenia reguły odgrywają rolę porządkującą i koordynującą. Umożliwiają bowiem wybór rodzaju operacji i określają kolejność ich wykonywania. Mogą mieć charakter heurystyczny lub algorytmiczny. Ponieważ reguły, łącząc się w systemy, tworzą określone strategie myślenia, w związku z tym mówi się na ogół o heurystycznych i algorytmicznych metodach myślenia.

⁹ Rzecz jasna, spostrzeganie nie stanowi warunku koniecznego tworzenia wyobrażeń. Potrafię bowiem utworzyć w swoim umyśle wizerunek przedmiotu, którego nigdy wcześniej nie widziałem. Wyobrażenia polegają więc na tworzeniu bądź odtwarzaniu obrazów rzeczy i zjawisk, ale również na ich przekształcaniu.

3.3. Metody myślenia

Współczesna psychologia uczenia się rozróżnia dwa rodzaje myślenia – produktywne i nieproduktywne. Rezultatem myślenia produktywnego jest wytworzenie informacji. Myślenie nieproduktywne polega natomiast na stosowaniu przez podmiot wiedzy już utrwalonej (Kozielecki, 1969).

Podstawę klasyfikacji czynności intelektualnych stanowią rodzaje zadań oraz sposoby ich rozwiązywania. Ze względu na typ zadań, jakie stoją przed podmiotem, można wyodrębnić zadania, których rozwiązanie będzie wymagać wytwarzania i wyboru informacji, oraz zadania, które wymagają jedynie wyboru informacji. Myślenie jest tym rodzajem czynności, który wiąże się z rozwiązywaniem zadań pewnego rodzaju (tj. takich zadań, które wymagają od podmiotu wyboru i wytwarzania informacji). Należy przy tym pamiętać, że myślenie występuje jedynie wówczas, gdy podmiot ma do czynienia z sytuacją problemową.

Każdy człowiek posiada dwa względnie wyodrębnione systemy myślowe, które funkcjonują w sytuacjach problemowych: generator pomysłów, który wytwarza różne hipotezy i pomysły, oraz ewaluator pomysłów (system sprawdzający), który jest odpowiedzialny za dokonanie wyboru oraz ocenę tych hipotez i pomysłów¹⁰.

Ze względu na istnienie dwóch systemów myślowych wyodrębnia się z kolei dwa rodzaje problemów rozwiązywanych przez człowieka (Such, Szcześniak, 1999) – problemy typu „odkryć” oraz „skonstruować”. Odkryć można jakieś zjawisko, relację, zależność (przyczynową, funkcjonalną) lub prawidłowość, które występują w rzeczywistości. Pomysłami rozwiązania, które występują w tej klasie problemów, są hipotezy, czyli domniemane odkrycia. Wynaleźć można natomiast jedynie to, czego dotąd nie było (wynaleźć to wytworzyć coś nowego). Rozwiązania problemów typu „odkryć” oceniamy pod względem ich prawdziwości, czyli zgodności hipotez z rzeczywistością. Natomiast rozwiązania problemów typu „skonstruować” oceniamy pod względem ich skuteczności, prostoty i oryginalności¹¹.

¹⁰ Rozróżnienie dwóch systemów myślowych zakłada analogię między myśleniem ludzkim a maszynowym. W podobny sposób działają bowiem urządzenia cybernetyczne, które z jednej strony wytwarzają odpowiednie ruchy, a z drugiej strony dokonują selekcji poszczególnych posunięć według określonych kryteriów.

¹¹ Dualizm odkrycia i konstrukcji jest znany w nauce jako antynomia pomiędzy kontekstem odkrycia (nie poddającym się logicznej analizie) a kontekstem uzasadnienia (Reichenbach, 1960). Zagadnienie to często formułuje się w następujący sposób: jak genialną intuicję zakomunikować za pomocą języka przedmiotowego?

3.3.1. Heurystyki

Wytworzyć informację to znaleźć pomysł rozwiązania danego zadania oraz sposoby, które pozwolą na sprawdzenie tego pomysłu (ocena, wybór i wzajemne przyporządkowanie pomysłu rozwiązania zadania i sposobu jego sprawdzenia).

Proces wytwarzania pomysłów jest procesem heurystycznym (gr. *heurisco* – sztuka odkrywania nowych rzeczy). Myślenie w procesie heurystycznym przebiega według reguł i metod, które sprzyjają wytwarzaniu pomysłów, lecz nie gwarantują pozytywnego rezultatu (Kozielecki, 1969). Heurystyki obejmują zatem wszelkie zasady, wskazówki i taktyki, które regulują przebieg procesu poszukiwania rozwiązania, lecz nie zapewniają pożądanego wyniku (np. metoda Osborna). Innymi słowy: heurystyki to zawodne reguły wytwarzania pomysłów, stanowiące schematy racjonalizujące poszukiwanie rozwiązań problemów (Altszuller, 1975).

Proces heurystyczny polega na wytwarzaniu informacji i stawianiu hipotez. Proces wytwarzania pomysłów ma jednocześnie wiele cech, które różnią go od procesów rozumowania, wyboru, oceny oraz sprawdzania:

1. proces heurystyczny nie przebiega zgodnie z zasadami logiki – nie jest kierowany przez reguły logiczne, które odznaczają się niezawodnością (ale też nie musi im przeczyć)¹²;

2. proces heurystyczny jest w wysokim stopniu intuicyjny (tj. kierowany w znacznej mierze przez wycucie problemu i uchwycenie sytuacji problemowej). Stąd przekonanie niektórych badaczy, że proces wytwarzania pomysłów nigdy nie zostanie zbadany, ponieważ reguły, według których on przebiega, nie mogą być poznane.

Każdy człowiek dysponuje całą rodziną heurystyk (mniej lub bardziej racjonalnych, czasami nawet wzajemnie sprzecznych). Niektóre heurystyki funkcjonują nieświadomie – człowiek może nie zdawać sobie sprawy z tego, że ukierunkowują one jego poszukiwania. Człowiek przyswaja sobie nowe reguły heurystyczne (w miarę uczenia się oraz rozwiązywania coraz to nowych problemów), natomiast część heurystyk jest spychana w podświadomość. To, jaki system heurystyczny znajdzie w danym wypadku zastosowanie, zależy od struktury problemu, celu (w jakim jest on rozwiązywany) i zdolności intelektualnych człowieka (Such, Szcześniak, 1999).

¹² Badania psychologiczne dowiodły, że zdolności do wysuwania pomysłów są ujemnie skorelowane ze zdolnością przeprowadzania rozumowań (w szczególności ścisłych wnioskowań dedukcyjnych, które przebiegają zgodnie z regułami logiki).

3.3.2. Algorytmy

Przeciwieństwem metod heurystycznych są metody algorytmiczne, które odgrywają zasadniczą rolę w myśleniu logicznym (krytycznym, analitycznym). Pomiędzy regułami heurystycznymi i algorytmicznymi zachodzą istotne różnice.

Proces wytwarzania hipotez nie zawsze prowadzi do powstania nowych informacji. Nawet najlepsze taktyki heurystyczne nie gwarantują bowiem, że ich zastosowanie doprowadzi do rozwiązania problemu. Oczywiście, poszczególne reguły heurystyczne mogą się znacznie różnić od siebie stopniem niezawodności, skuteczności oraz racjonalności. Niełatwo jest jednak ustalić, które z nich są lepsze, a które gorsze. W przypadku reguł heurystycznych mamy zawsze do czynienia z taktyką odkrywania i strategią poszukiwań. Zawodność jest zatem główną cechą, która różni heurystyki od algorytmów (tj. operacji niezawodnych, zawsze wiodących do sukcesu).

Heurystyki, w przeciwieństwie do algorytmów, nie są podane w postaci listy ściśle określonych, elementarnych operacji nad znakami, bądź też operacji myślowych, które należy wedle określonej kolejności wykonać, aby rozwiązać problem.

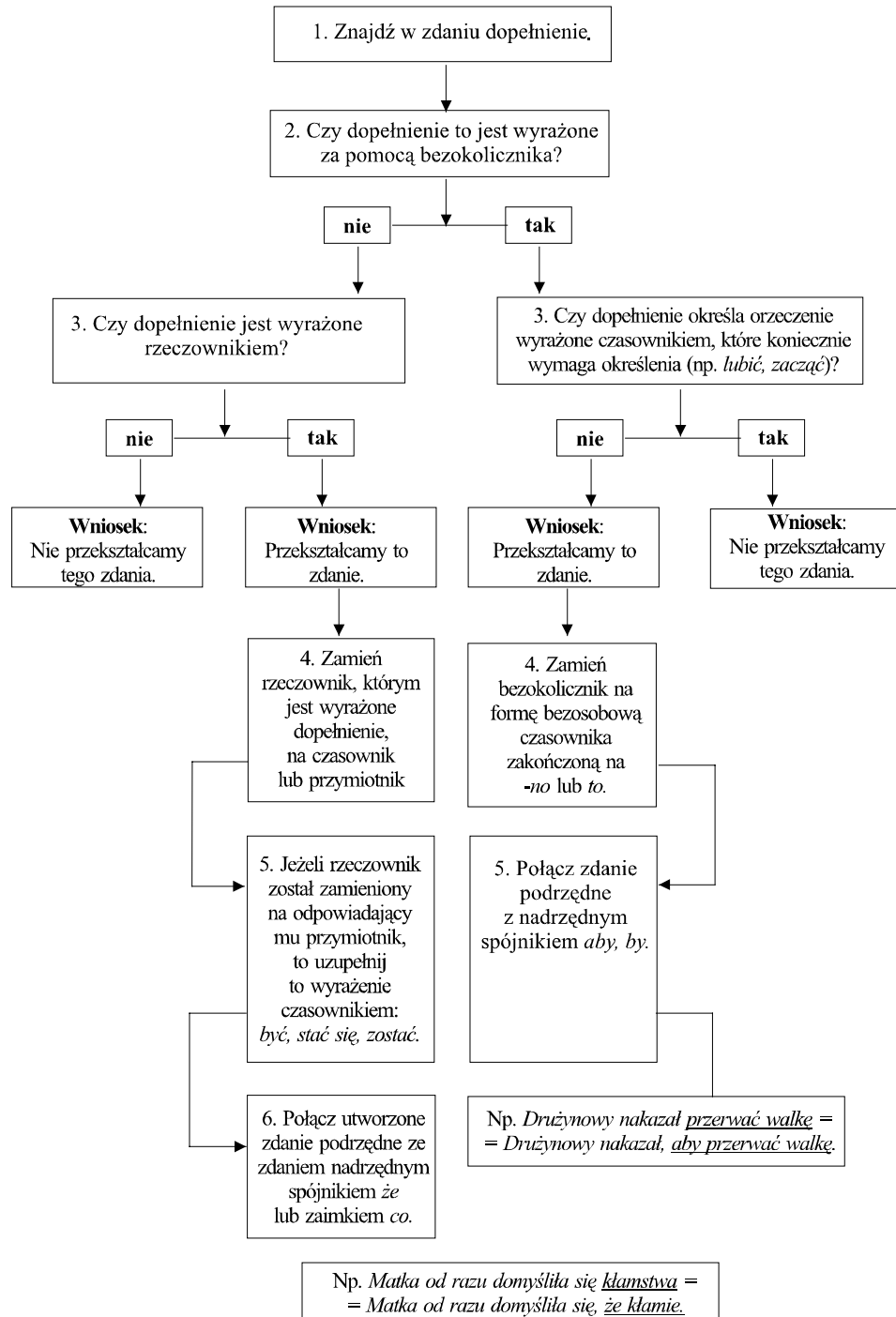
3.4. Praktyczne wykorzystanie odkryć psychologii w konstruowaniu algorytmów

Chciałbym teraz odnieść wszystko to, co dotąd zostało powiedziane na temat algorytmu jako metody myślenia, do zaproponowanej przeze mnie metody algorytmizacji wybranych działów gramatyki i ortografii.

Myślenie algorytmiczne polega na operowaniu pojęciami w oderwaniu od konkretnych cech przedmiotów bądź zdarzeń. Stanowi ono zatem przejaw myślenia pojęciowego. W trakcie przetwarzania algorytmicznego uruchamiane są niemal wszystkie dostępne człowiekowi operacje pojęciowe – z porównywaniem i abstrahowaniem włącznie. Jednak podstawowe operacje, z jakimi mamy do czynienia w myśleniu algorytmicznym to analiza oraz synteza.

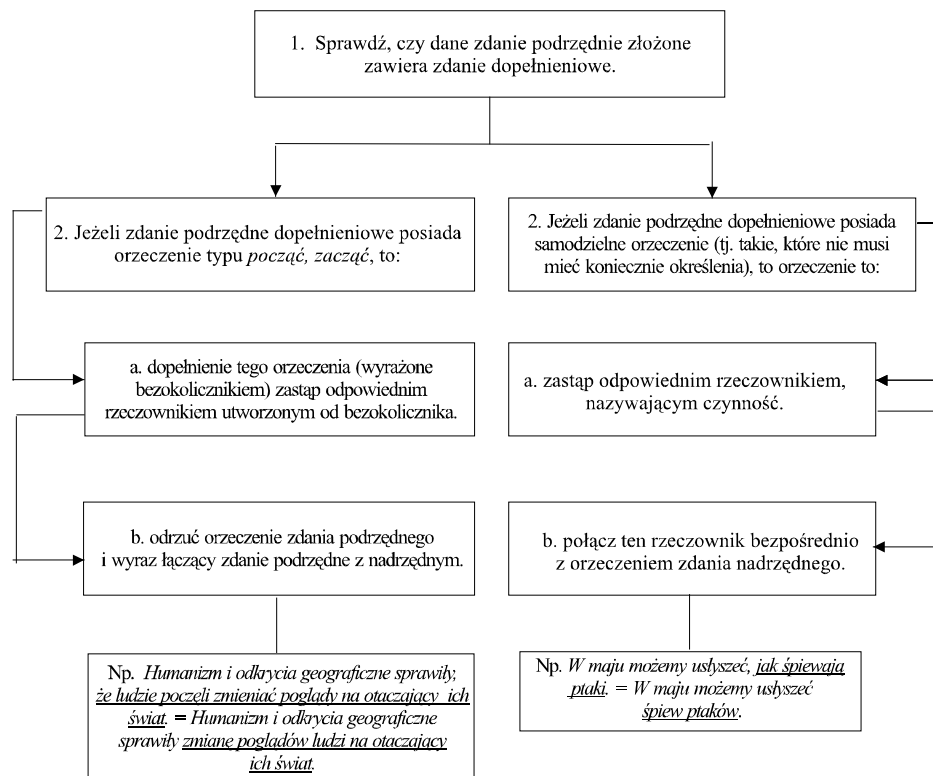
Każdy algorytm może działać dwukierunkowo: albo składając z elementów jakąś całość poprzez układ operacji wykonywanych na tych elementach (synteza), albo poszukując elementów jakiejś całości lub struktury czynności, która prowadzi do jej złożenia (analiza).

W pierwszym przypadku algorytm określa dane pojęcie lub obiekt przez podanie operacji, które prowadzą do jego utworzenia. W takim ujęciu algorytm pełni funkcję efektywnej definicji operacyjnej, por. algorytm przekształcania zdań pojedynczych z dopełnieniem na zdania podrzędnie złożone z podrzędnym zdaniem dopełnieniowym, który wyszczególnia obiekty (m.in.



dopełnienie wyrażone rzeczownikiem i bezokolicznikiem) i operacje (1, 2, 3, 4, 5, 6) prowadzące do utworzenia zdania podrzędnie złożonego z podrzędnym zdaniem dopełnieniowym.

W drugim przypadku algorytm umożliwia poznanie struktury operacji, które prowadzą do powstania jakiejś całości, por. algorytm przekształcania zdań podrzędnie złożonych z podrzędnym zdaniem dopełnieniowym na zdania pojedyncze z dopełnieniem, który ukazuje, jakie operacje (1, 2, 2a, 2b) i obiekty (m.in. zdanie podrzędne dopełnieniowe, orzeczenie wymagające dopełnienia w bezokoliczniku oraz orzeczenie dające się zastąpić rzeczownikiem nazywającym czynność) wchodzi w obręb struktury zdania podrzędnie złożonego z podrzędnym zdaniem dopełnieniowym.



Jak już nadmieniałem, myślenie algorytmiczne polega m.in. na operowaniu pojęciami w oderwaniu od konkretnych cech przedmiotów bądź zdarzeń. Operacje algorytmiczne przeprowadzane są na obiektach unilateralnych (napi-sy, kształty, „ślady atramentu na papierze”) oraz na obiektach, które stanowią

rezultat obróbki intelektualno-pojęciowej (pojęcia i reguły gramatyczne). Operacje algorytmiczne zasadzają się z reguły na skorelowaniu typu i okazji określonego terminu gramatycznego. Pytanie algorytmiczne odnosi bowiem konkretne wyrażenie do określonej klasy abstrakcji, czyli poprzez okaz (argument: „Wyraz, który masz napisać”) wskazuje na typ (predykat: „jest rzeczownikiem”)¹³.

Operacje algorytmiczne to działania intelektualne. Weźmy operację algorytmiczną, którą reprezentuje pytanie: *Czy litera ó wymienia się z samogłoską o lub e w innych formach gramatycznych danego wyrazu lub w wyrazach pokrewnych?* Operacja algorytmiczna polega tutaj na:

1. rozpoznaniu danego wyrazu (kształtu, napisu) jako sprawiającego trudność w pisowni: *nióśł* lub *niust*;
2. wstawieniu danego wyrazu w strukturę algorytmu;
3. utworzeniu z danego wyrazu innych jego form gramatycznych: *nióśł* – *niose* – *niesie*;
4. zestawieniu i porównaniu danego wyrazu z innymi jego formami gramatycznymi: *nióśł* : *niose* : *niesie*;
5. uzasadnieniu pisowni litery *ó* w danym wyrazie ze względu na określoną regułę ortograficzną (w tym przypadku: historyczna zasada pisowni, oparta na tzw. przegłosie polskim).

3.5. Nauczanie algorytmiczne jako metoda dydaktyczna

Jest metoda, która w procesie dydaktycznym wykorzystuje algorytmy. Algorytmizacja nauczania stanowi odmianę nauczania programowanego (choć powstała w opozycji do tradycyjnych programów: liniowego i rozgałęzionego). Pomiędzy algorytmizacją nauczania a klasycznym nauczaniem programowym zachodzą jednak istotne różnice. Nauczanie programowane służy przede wszystkim zdobywaniu określonych wiadomości, podczas gdy algorytmizacja nauczania jest bardziej nastawiona na kształcenie określonych umiejętności i nawyków (aż do ich automatyzacji).

Literatura dostarcza wielu przykładów prób zdefiniowania pojęcia algorytmu dla potrzeb dydaktyki. Pomysł szerszego zastosowania algorytmów w procesie nauczania zaproponował rosyjski uczonec L. N. Łanda. Jego zdaniem, algorytm to „przepis, który określa, jaki skończony ciąg operacji należy kolejno wykonać, aby rozwiązać wszystkie zadania danej klasy” (Łanda,

¹³ Zagadnienie pytań i poleceń algorytmicznych zostało szczegółowo omówione w: Nowak, 2002a,b.

1961). Zacytowana definicja eksponuje istotne cechy algorytmu (Kozielecki, 1969), rozumianego jako procedura dydaktyczna.

1. Określoność. Każdy algorytm wskazuje jednoznacznie, jaki skończony ciąg operacji należy wykonać, aby rozwiązać dane zadanie. Przepisy, które tworzą algorytm, muszą więc być ściśle określone i nie mogą dopuszczać żadnych dowolności w swojej interpretacji¹⁴.

2. Masowość. Zastosowanie algorytmu nie ogranicza się do pojedynczego, szczegółowego przypadku, lecz odnosi się do pewnej klasy zadań, np. algorytm pisowni wyrazów z „ó” i „u” obejmuje wszystkie wyrazy języka polskiego i nie ma w tym zbiorze takich słowoform, do których algorytm ten nie mógłby być zastosowany.

3. Skuteczność. Algorytm (jako przepis na wykonanie jakiegoś zadania) jest skierowany na uzyskanie określonego wyniku, przy czym wynik ten jest zawsze osiągnięty (możliwie najkrótszą drogą).

4. Determinizm. Algorytmiczne rozwiązywanie zadań jest działalnością intelektualną, która jest ściśle ukierunkowana, zaprogramowana i nie dopuszcza żadnej dowolności¹⁵.

4.2. Rozwiązywanie zadań z wykorzystaniem algorytmów

Uczenie się człowieka, rozpatrywane zarówno na płaszczyźnie filogenetycznej, jak i ontogenetycznej, polega na rozwiązywaniu problemów (zadań). Zadania, jakie można postawić przed człowiekiem, ograniczają się do problemów, które dają się rozwiązać za pomocą algorytmów oraz problemów, które nie dają się rozwiązać za pomocą algorytmów (lecz jedynie przez mniej lub bardziej dokładne reguły heurystyczne).

W obrębie moich zainteresowań sytuują się te zadania, które można rozwiązać za pomocą algorytmów. Algorytmiczne (zdyscyplinowane) rozwiązywanie zadań prowadzi przez następujące etapy:

1. postawienie problemu – sformułowanie zadania;

¹⁴ Rzecz jasna, rozkład danej czynności na operacje elementarne jest zależny od tego, dla kogo lub czego jest przeznaczony dany przepis algorytmiczny. Inaczej będą formułowane przepisy dla maszyn i urzędzeń, a inaczej dla człowieka. Inaczej będzie brzmiał algorytm przeznaczony dla językoznawcy, a inaczej dla ucznia gimnazjum. Kwestią otwartą jest to, czy dla wykonania zadania przez człowieka tworzy się algorytmy, czy też jedynie przepisy dla danej czynności, zbliżone w swoim charakterze do algorytmów.

¹⁵ Algorytmiczne rozwiązywanie zadań jest zatem takim procesem, który może być powtórzony przez dowolną osobę lub maszynę (jeżeli algorytm jest włożony do maszyny) i prowadzi zawsze przy jednakowych danych do jednakowych rezultatów.

2. specyfikacja zadania – szukanie związku między danymi wejściowymi a wyjściowymi:

- a. określenie danych wejściowych – sprecyzowanie wartości, które są dane,
- b. określenie danych wyjściowych – sprecyzowanie wartości, które trzeba znaleźć;

3. poszukiwanie metody rozwiązania – wypisanie wzorów, które łączą dane wejściowe z wyjściowymi lub poszukiwanie twierdzenia, które można zastosować;

4. skonstruowanie schematu postępowania w postaci algorytmu;

5. opis rozwiązania w możliwie jasny sposób – przedstawienie algorytmu w postaci opisu słownego, listy kroków bądź schematu blokowego;

6. analiza poprawności rozumowania;

7. testowanie algorytmu dla różnych danych – ocena efektywności przyjętej metody.

Jak już nadmieniałem, w procesie myślenia człowiek może posługiwać się zarówno regułami heurystycznymi, jak i algorytmicznymi. Podczas rozwiązywania zadań z wykorzystaniem algorytmów mogą się więc pojawić następujące sytuacje dydaktyczne (Kwiatkowski, 1982):

1. zadanie → algorytm → rozwiązanie;

2. zadanie → przekształcenie zadania → algorytm → rozwiązanie;

3. zadanie → metody heurystyczne → rozwiązanie → uporządkowanie procedury → algorytm.

Sytuacja 1. sprowadza się do tego, że uczeń zna i stosuje algorytm rozwiązania zadania. Sytuacja 2. polega na tym, że uczeń nie zna algorytmu, za pomocą którego można rozwiązać dane zadanie. Uczeń przekształca zatem zadanie do takiej postaci, że możliwe staje się zastosowanie znanego mu algorytmu. Sytuacja 3. zakłada, że rozwiązanie zadania, osiągnięte drogą stosowania reguł heurystycznych, może stać się algorytmem (po dokonaniu niezbędnych uściśleń).

Nauczanie za pomocą algorytmów może się zatem odbywać dwutorowo: poprzez podawanie uczniom gotowych algorytmów (por. sytuacja 1.), bądź poprzez układanie algorytmów przez uczniów pod kierunkiem nauczyciela (por. sytuacja 2. i 3.). Algorytmy (np. w postaci schematów blokowych) uczniowie mogą rysować w zeszycie lub przygotowywać w dowolnym edytorze graficznym czy tekstowym. Dowiedziono bowiem, że ci uczniowie, którzy współuczestniczą w tworzeniu algorytmów, bardziej trwale opanowują wiedzę, aby następnie w sposób operatywny móc posługiwać się nią podczas rozwiązywania zadań.

4.3. Efektywność nauczania za pomocą algorytmów

Efektywność nauczania algorytmicznego zależy od uwzględnienia trzech czynników (Porayski-Pomsta, 1981):

- sposobu opracowania algorytmu oraz jego wykorzystania podczas lekcji;
- budowy algorytmu;
- stanu wiedzy uczniów przed wprowadzeniem algorytmu.

Opracowanie algorytmu polega przede wszystkim na rozdzieleniu danej czynności na operacje elementarne oraz na rozłożeniu wyniku końcowego na wyniki częściowe, będące produktem poszczególnych, możliwie prostych, operacji. Kładzie się przy tym nacisk na to, aby algorytm był sformułowany w sposób zrozumiały dla uczniów. Algorytm musi wreszcie prowadzić we wszystkich przypadkach, w skończonej liczbie kroków, do poprawnego wyniku.

Algorytmy powinny posiadać budowę odpowiadającą budowie analizowanych zjawisk językowych. Konstruując algorytmy, należy uwzględniać jedynie istotne cechy zjawisk językowych. Jak zatem widać, już sama budowa algorytmu jest wielce pouczająca – zmusza bowiem ucznia do systematycznej i uporządkowanej obserwacji analizowanych zjawisk językowych. W ten sposób algorytmy uczą dostrzegania elementów istotnych, a także wyrabiają u ucznia technikę analizy i syntezy¹⁶.

Dowiedziano również, że efektywność nauczania z wykorzystaniem algorytmów pozostaje w ścisłym związku ze zgromadzoną wcześniej przez ucznia wiedzą (Porayski-Pomsta, 1981). Uczeń, który nie opanował dostatecznie wiadomości, nie może wykorzystać algorytmu, ponieważ nie rozumie zawartych w nim poleceń. Dlatego porządek zajęć powinien przewidywać omówienie reguł gramatycznych bądź ortograficznych, a następnie prezentację algorytmu.

W początkowej fazie badań nad wykorzystaniem algorytmów w procesie dydaktycznym ograniczono się do nauki gramatyki i elementów matematyki, a więc treści z natury swej algorytmicznych (Kwiatkowski 1982). Wyniki tych badań wykazały, że kształtowanie umiejętności rozwiązywania zadań matematycznych oraz umiejętności gramatycznych przy wykorzystaniu reguł

¹⁶ Czytanie algorytmu w kierunku od skrzynki wejścia do skrzynki wyjścia ukazuje procedurę rozwiązywania określonego zadania – możliwie najkrótszą i najdogodniejszą drogę do osiągnięcia pożądanego rezultatu. Natomiast algorytm czytany w odwrotnym porządku ukazuje gramatyczną strukturę języka. Próba odczytywania algorytmu wstecz pozwala na odtworzenie reguł gramatycznych, jakimi rządzi się wybrany wycinek systemu językowego. Algorytm jako narzędzie dydaktyczne może zatem służyć podczas utrwalania i ćwiczenia nabytego uprzednio materiału. Każdy algorytm równie dobrze może zostać wykorzystany do odtwarzania gramatyki danego języka na podstawie korpusu tekstowego. Czytanie algorytmu przypomina nieco analizę związania i rozwijania, czyli przekładania porządku linearnego na strukturalny (i na odwrot).

algorytmicznych daje lepsze efekty w stosunku do nauczania i uczenia tradycyjnymi metodami (Porayski-Pomsta, 1978/1979).

Zdaniem badaczy (Polański, 1985, 1995), nie w każdym jednak dziale nauczania języka polskiego można stosować algorytmizację jako metodę dydaktyczną. Algorytmy można z powodzeniem proponować w nauczaniu składni lub morfologii (jako wprowadzenie nowego materiału bądź jego utrwalenie). „Trudna do zalgorytmizowania jest natomiast polska ortografia, ponieważ wśród reguł przeważają takie, które mają dużą liczbę wyjątków. Należy zatem algorytmizować tylko te reguły, od których nie ma (lub jest mała liczba) wyjątków. Duża liczba wyjątków może bowiem zamazać czytelność, a co a tym idzie, utrudnić opanowanie reguły ortograficznej” (Polański, 1995). Dlatego też w literaturze na ogół brak jest przykładów zastosowania metod algorytmicznych w nauczaniu ortografii¹⁷.

ZAKOŃCZENIE

Jestem przekonany zarówno o potrzebie kontynuowania badań w zakresie algorytmizacji dydaktyki, jak i o korzyściach, jakie płyną ze stosowania algorytmów w procesie nauczania języka polskiego. Sprzyja temu program nauczania, który zakłada blokowy i zintegrowany charakter kształcenia. Sytuacja taka umożliwia łączenie w procesie dydaktycznym szeregu treści z pogranicza matematyki, informatyki oraz gramatyki języka polskiego. Tym bardziej, że nowe programy nauczania przewidują kształcenie uczniów w zakresie specyfikacji zadań i sporządzania algorytmów (np. na lekcjach informatyki w gimnazjum). Zarówno blokowy charakter nauczania, jak i postępująca komputeryzacja otaczającej nas rzeczywistości sprzyjają nauczaniu algorytmicznemu i sprawiają, że czyni ono zadość potrzebom dydaktyczno-naukowym XXI wieku.

Wydaje mi się, że zdołałem jedynie zasygnalizować kilka ważniejszych zagadnień. Szereg problemów, jakie zdołałem poruszyć, domaga się gruntowniejszego zbadania oraz większej kompetencji badawczej.

¹⁷ Tę lukę wydawniczą wypełniają pozycje: T. N o w a k, *Ortografia, co w szóstkę trafia*, Warszawa 2002, oraz t e n ż e, *Powtórka z gramatyki. Gimnazjum*, Warszawa 2004. Pochodzi stąd większość z przytoczonych w tym artykule algorytmów.

BIBLIOGRAFIA

- A j d u k i e w i c z K., 1985: O spójności syntaktycznej, w: *Język i poznanie*, t. I, Warszawa.
- A l t s z u l l e r H., 1975: *Algorytm wynalazku*, Warszawa.
- A ł f i e r o w a Z., 1977: *Teoria algorytmów*, Warszawa.
- A p r e s j a n J. D., 1971: *Koncepcje i metody współczesnej lingwistyki strukturalnej*, Warszawa.
- B a ń k o M., 1990: Niektóre problemy oceny adekwatności gramatyki (na podstawie fragmentu gramatyki Szpakowicza), w: *Studia gramatyczne*, t. IX.
- B i a ł y n i c k a - B i r u l a I., B i a ł y n i c k i - B i r u l a I., 2002: *Modelowanie rzeczywistości. Od gry w życie Conwaya przez żuka Mandelbrota do maszyny Turinga*, Warszawa.
- B i e ń J. S., 1993: Komputerowa weryfikacja formalnej gramatyki, „*Biuletyn PTJ*”, t. LII, Warszawa.
- B l i k l e A., 1971: *Automaty i gramatyki. Wstęp do lingwistyki matematycznej*, Warszawa.
- B o r k o w s k i L., 1991: *Wprowadzenie do logiki i teorii mnogości*, Lublin.
- B u s z k o w s k i W., 1999: *Składnia*, w: K. A j d u k i e w i c z, *Teoria znaczenia*, red. R. Wójcicki, Warszawa.
- F o n t a ń s k i H., C h o m i a k M., 2000: *Gramatyka języka łemkowskiego*, Katowice.
- G r z e g o r c z y k o w a R., 2001: *Wprowadzenie do semantyki językoznawczej*, Warszawa.
- H e l l e r M., L u b a ń s k i M., Ś l a g a Sz. W., 1980: *Zagadnienia filozoficzne współczesnej nauki. Wstęp do filozofii przyrody*, Warszawa.
- H o d g e s A., 1998: *Turing. Miniatury filozoficzne*, Warszawa.
- K o z i e l e c k i J., 1969: *Rozwiązywanie problemów*, Warszawa.
- K r a j e w s k i S., 2003: *Twierdzenie Gödla i jego interpretacje filozoficzne*, Warszawa.
- K u n z m a n n P., B u r k a r d P., W i e d m a n n F., 1999: *Atlas filozofii*, Warszawa.
- K w i a t k o w s k i S., 1982: *Heurystyki i algorytmy w procesie dydaktycznym*, Warszawa.
- Logika formalna. Zarys encyklopedyczny*, 1987, red. W. Marciszewski, Warszawa.
- L o r e n z K. Z., 1977: *Odwrotna strona zwierciadła. Próba historii naturalnej ludzkiego poznania*, Warszawa.
- Ł a n d a L. N., 1961: *Nauka metod racjonalnego myślenia a problem rozwiązywania algorytmów*, „*Nowa szkoła*”, nr 11.
- M a r c i s z e w s k i W., red., 1987: *Logika formalna. Zarys encyklopedyczny*, Warszawa.
- M u r a w s k i R., 1995: *Filozofia matematyki. Zarys dziejów*, Warszawa.
- N o w a k T., 2002: *Algorytmiczne nauczanie gramatyki – teoria i wnioski dla praktyki*, w: *Szczególne problemy człowieka i edukacji u progu XXI wieku. Księga jubileuszowa ofiarowana Profesorowi Bogdanowi Snochowi*, Częstochowa.
- N o w a k T., 2002b: *Algorytmiczne nauczanie składni – teoria i wnioski dla praktyki*, w: *Studencki ruch naukowy – wyzwania XXI wieku. Materiały Międzynarodowej Konferencji Studenckich Kół Naukowych*, Siedlce.
- P e n r o s e R., 1996: *Nowy umysł cesarza, o komputerach, umyśle i prawach fizyki*, Warszawa.
- P o l a ń s k i E., 1985: *Kształcenie językowe w klasach 4-8. Poradnik metodyczny*, Warszawa.
- P o l a ń s k i E., 1995: *Dydaktyka ortografii i interpunkcji*, Warszawa.
- P o p p e r K. P., 1992: *Wiedza obiektywna. Ewolucyjna teoria epistemologiczna*, Warszawa.

- P o r a y s k i - P o m s t a J., 1978/1979: Zastosowanie metody algorytmów w nauczaniu składni w klasie 6, „Język Polski w Szkole”, z. 2.
- P o r a y s k i - P o m s t a J., 1981: Nauczanie algorytmiczne gramatyki języka polskiego w szkole. Podstawy psychologiczne i językoznawcze, „Poradnik Językowy”, z. 5.
- R e i c h e n b a c h H., 1960: Powstanie filozofii naukowej, Warszawa.
- S e m e n i u k - P o l k o w s k a M., P o l k o w s k i L. T., 1994: Matematyka dla humanistów, Warszawa.
- S ł o m k i e w i c z S., 1972: Nauczanie algorytmiczne a psychologiczna teoria czynności, Warszawa.
- S u c h J., S z c z e ś n i a k M., 1999: Filozofia nauki, Poznań.
- S u r a j Z., R u m a k T., 1995: Algorytmiczne rozwiązywanie zadań i problemów, Rzeszów.
- W i c k l e r W., 1991: Biologia dziesięciu przykazań, Warszawa.

ALGORITHMISATION OF GRAMMAR AND ORTHOGRAPHY –
THE THEORY AND ITS APPLICATION

S u m m a r y

The paper develops the problem of application of the theory of mathematical algorithms in the linguistics and didactics of language. The logical, linguistic and psychological theoretical knowledge is exemplified by the author by many instances algorithmics.

Słowa kluczowe: algorytm, modelowanie, język, myślenie, zadanie.

Key words: algorithm, modelling, language, thinking, enterprise.