

Wojciech Jawień

Akademia Medyczna im. Mikołaja Kopernika

w Krakowie

**Koncepcja komputerowego systemu informacji o leku  
posiadającego cechy systemu ekspertowego  
w zakresie leków krążeniowych.**

praca doktorska wykonana w Zakładzie Chemii Fizycznej

Promotor: Prof. dr hab. Adam Danek

Prof. dr hab. Jerzy Brandys

Bibl. Medyczna CM UJ



1816096165

Kraków, 1989.

Możliwość zrealizowania tej pracy zawdzięczam wielu osobom, którym pragnę w tym miejscu gorąco podziękować:

Pan prof. dr hab. Adam Danek, dzięki swemu ogromnemu entuzjazmowi dla wykorzystania komputerów w farmacji klinicznej, stworzył warunki, w których można było podjąć badania nad komputerowym systemem informacji o leku. Niestety, nie zdążył On zapoznać się z otrzymanymi rezultatami pracy, którą zainicjował i której sumiennie doglądał.

Niełatwej roli kontynuowania opieki nad pracą podjął się Pan prof. dr hab. Jerzy Brandys.

Pani dr Joanna Szymura-Oleksiak pomogła usunąć liczne usterki rękopisu.

Nieocenioną pomoc w przezwyciężaniu różnych trudności wielokrotnie ofiarował Pan doc. dr hab. Jerzy Porębski, Dziekan Wydziału Farmaceutycznego AM.

Sfinalizowanie pracy nie byłoby możliwe, gdyby nie życzliwość Dyrektora Instytutu Farmakologii PAN w Krakowie, Pana prof. dr hab. Jerzego Maja, który wyraził zgodę na korzystanie z komputera IBM PC w Instytucie. Ogromne są moje zobowiązania wobec Państwa dr Krys-tyny Gołembiewskiej, dr Anny Czyrak i mgra Jerzego Superaty z Instytutu Farmakologii, którzy z podziwu godną wyrozumiałością przez ponad rok znosili trudy związane z pracą na komputerze w ich pracowni.

Dotarcie do wielu publikacji z zakresu sztucznej inteligencji zawdzięczam Panu inż. Wiesławowi Golcowi z Zakładu Farmakologii Śląskiej Akademii Medycznej w Katowicach.

Wreszcie, *last but not least*, wielki jest mój dług wobec Żony Marii, która była dla mnie w okresie pisania pracy nie tylko współpracownikiem, lecz przede wszystkim cierpliwym Przyjacielem. Tylko dzięki Jej nieustannej pomocy znalazłem dość wytrwałości, by pracę tę doprowadzić do końca.

Wojciech Jawień



## SPIS TREŚCI

0.	Wstęp.	1
0.1	Stan badań w kraju.	3
0.2	Cel pracy.	4
0.3	Realizacja.	6
0.4	Podział tekstu.	7
1.	Bazy danych - wprowadzenie.	8
1.1	Podstawowe pojęcia.	8
1.2	Modele danych.	10
1.3	Relacyjny model danych.	12
1.3.1.	Klucze relacji.	14
1.3.2.	Schemat logiczny.	14
1.3.3.	Postaci normalne relacji.	15
1.3.4.	Symetryczne składniki klucza.	22
1.3.5.	Operacje relacyjne.	23
1.3.6.	Języki zapytań do bazy danych.	24
2.	Wprowadzenie do systemów doradczych.	26
2.1.	Systemy doradcze jako kierunek badań nad sztuczną inteligencją.	26
2.2.	Podstawowe definicje.	27
2.3.	Reprezentacja wiedzy.	28
2.4.	Strategie wnioskowania.	31
2.5.	Pozyskiwanie wiedzy.	33
2.6.	Rozumowanie przybliżone.	33
2.6.1.	Zbiory rozmyte.	34
2.6.2.	Zastosowanie w automatycznym wnioskowaniu.	35
3.	Projekt systemu.	37
3.1	Ogólne założenia projektu.	37
3.2	Wytyczne o charakterze informatycznym.	37
3.2.1.	Uzasadnienie.	38
3.3.	Założenia projektu Systemu Zarządzania Baza Danych.	43
3.3.1.	Projekt języka zapytań.	45
3.4.	Założenia projektu programu pozwalającego na tworzenie systemów doradczych.	48
3.4.1.	Uzasadnienie.	48
3.4.2.	Projekt reprezentacji stwierdzeń.	49
3.4.3.	Wykorzystanie środków SZBD do reprezentacji bazy wiedzy.	50
3.5.	Projekt bazy danych o lekach krażeńiowych.	52
3.5.1.	Schemat logiczny.	52
3.5.2.	Organizacja fizyczna.	55
3.6.	Projekt systemu doradczego dla doboru leku krażeńiowego.	57
3.6.1.	Schemat logiczny relacji WYWIAD.	57
3.6.2.	Organizacja fizyczna.	60
3.6.3.	Cele (etapy) wnioskowania.	60

4.	Realizacja programu	63	
4.1.	Wybór języka programowania.	63	
4.2.	Struktury danych.	67	
4.2.1.	Słowniki i identyfikatory.	67	
4.2.2.	Reprezentacja struktury bazy danych.	68	
4.2.3.	Informacja tekstowa.	69	
4.2.4.	Wyrażenia arytmetyczne i zapytania do bazy danych.		70
4.2.5.	Reprezentacja wiedzy.	71	
4.3.	Elementy systemu.	75	
4.3.1.	Moduły SZBD.	75	
4.3.1.1.	Edytory struktury danych.	75	
4.3.1.2.	Procedury podstawowe.	76	
4.3.1.3.	Moduły wprowadzania, prezentacji i wyszukiwania informacji.	76	
4.3.1.4.	Procedury zapisu i odczytu struktur bazy danych.		79
4.3.2.	Translator wyrażień arytmetyczno-logicznych.	80	
4.3.3.	Moduł dla tworzenia systemu doradczego.	83	
4.3.3.1.	Edytor reguł.	83	
4.3.4.	Moduł eksploatacji systemu doradczego.	85	
4.3.5.	Komunikacja z użytkownikiem.	88	
5.	Konstrukcja systemu.	90	
5.1.	Konstrukcja bazy danych.	90	
5.1.1.	Określenie schematu i organizacji fizycznej.		90
5.1.2.	Wprowadzanie danych.	92	
5.2.	Konstrukcja systemu doradczego.	94	
5.2.1.	Redagowanie bazy wiedzy.	94	
6.	Wyniki.	96	
6.1.	Przykłady ilustrujące działanie systemu.	96	
6.1.1.	Wyszukiwanie informacji.	96	
6.1.1.1.	Uzyskanie informacji o konkretnym leku.	96	
6.1.1.2.	Wyszukiwanie z wyborem interesujących atrybutów.		101
6.1.2.	Wybór leku przez system doradczy.	105	
6.1.2.1.	Wywiad lekarski.	105	
6.1.2.2.	Analiza przypadku.	105	
6.1.2.3.	Prezentacja i objaśnianie rozwiązań.	108	
6.2.	Ocena programu.	111	
6.2.1.	System zarządzania bazą danych.	111	
6.2.2.	Moduły tworzenia i eksploatacji systemu doradczego.		112
6.3.	Ocena systemu informacyjnego.	114	
6.4.	Ocena systemu doradczego.	114	
6.5.	Wnioski.	117	
	Przypisy.	121	
	Piśmiennictwo.	123	



## O. WSTĘP

Dokonywanie wyboru i rozpoznawanie obiektów są czynnościami stale obecnymi w działalności człowieka. Podjęcie właściwej decyzji w odpowiednim czasie niejednokrotnie przesądza o powodzeniu wielu przedsięwzięć - bez względu na to, czy realizowane są one na polu ekonomicznym, militarnym, czy medycznym.

Proces decyzyjny wymaga z jednej strony zgromadzenia stosownych informacji, z drugiej zaś odpowiedniej wiedzy, pozwalającej te informacje właściwie wykorzystać.

Efektywnym przechowywaniem i wyszukiwaniem informacji zajmuje się jedna z dziedzin informatyki - teoria baz danych [1,2], w której systematyzację duży wkład wniosły prace Codda [3,4].

Zagadnienia komputerowej reprezentacji wiedzy i automatycznego wnioskowania wchodzi w skład innej dyscypliny informatycznej, zajmującej się badaniami nad sztuczną inteligencją [5], a zwłaszcza jej działu dotyczącego tzw. systemów doradczych (ekspertowych [6,7]).

W działalności lekarza czy farmaceuty klinicznego zarówno sprawne uzyskanie informacji o leku, jak i właściwy, dostosowany do indywidualnych cech pacjenta dobór tego leku mają zasadnicze znaczenie i decydują o skuteczności farmakoterapii. Stworzenie komputerowego systemu ułatwiającego uzyskanie i wyzyskanie informacji o leku daje zatem szansę znacznego usprawnienia pracy lekarza bądź farmaceuty.

Na świecie istnieje obecnie wiele ośrodków rozpowszechniających różnego rodzaju informacje na temat leków; większość tych systemów informacyjnych posiada komputerowe bazy danych, udostępniane użytkownikom w trybie *on-line*, a ostatnio również na dyskach kompaktowych [8,9]. Do najbardziej znanych systemów tego typu należą: "Martindale-Online", "Drugdex", "Pharmline", "AMA-Drug Information

Base". Znane systemy bibliograficzne, takie jak "Excerpta Medica", "Chemical Abstracts" czy "Medline" uwzględniają zagadnienia związane z lekami. Wykorzystanie na szerszą skalę w naszym kraju tych znacznych zasobów informacji nie jest jednak obecnie możliwe z wielu oczywistych powodów. Ponadto wiele danych dotyczących leku zależy od kraju, w którym lek ten jest stosowany, dlatego też informacje te mogłyby być niewystarczające dla polskiego odbiorcy. Zachodzi zatem potrzeba opracowania odpowiedniego systemu informacyjnego w kraju, zwłaszcza że zapotrzebowanie na tego rodzaju informacje jest ogromne, czego dowodem może być choćby natychmiastowe rozchodzenie się kolejnych wydań i wznowień monografii Podlewskich [10]. Utworzenie takiego systemu, który powinien obejmować większość stosowanych w kraju środków farmaceutycznych, jest przedsięwzięciem złożonym, czasochłonnym, wymagającym zaangażowania znacznej liczby specjalistów, odpowiedniego zaplecza technicznego oraz finansowego. Nie jest to więc zadanie, którego mógłby się podjąć niewielki zespół zajmujący się w krakowskiej AM informacją o leku. Jednakże brak (na szczeblu centralnym) widocznych działań w kierunku stworzenia wspomnianego systemu z jednej strony i znaczne zainteresowanie środowiska problematyką informacyjną z drugiej, skłaniają do podjęcia rozwiązań alternatywnych. Jedną z możliwości jest szczegółowe opracowanie informacji o wąskiej grupie leków i udostępnienie jej w postaci systemu, który można eksploatować na coraz powszechniej dostępnych w placówkach Służby Zdrowia mikrokomputerach klasy IBM PC-XT/AT. Nawet pracując w małym zespole można, zajmując się niewielką grupą leków, zgromadzoną o nich wiedzę nie tylko wprowadzić do bazy danych (na ogół jako informację tekstową), lecz także usystematyzować ją, sformalizować i przedstawić w odpowiedniej postaci, tworząc tzw. bazę wiedzy [6,7,11]. Umożliwia to aktywne wykorzystanie tej wiedzy przez komputer dla ułatwienia użytkownikowi optymalnego doboru leku. System skonstruowany według

takich zasad ułatwiałby więc nie tylko sprawne odszukanie potrzebnych danych na temat konkretnego leku, lecz także spełniałby wobec lekarza rolę doradcy, pomagając w przeprowadzeniu wyczerpującej analizy wielu nieraz możliwych do zastosowania leków pod kątem ich przydatności w indywidualnym przypadku. Doświadczenia nabyte w takim zespole mogą być następnie wykorzystane przez inne, podobne ośrodki do opracowania innych grup leków; stwarza to możliwość wypracowania pewnego standardu, a nawet otwiera perspektywę późniejszej syntezy większego systemu, który wiązałby dotychczas niezależne podsystemy dotyczące leków z różnych grup.

#### 0.1 Stan badań w kraju.

Próby konstrukcji podobnych systemów informacji i systemów wspomagających dobór leku były już podejmowane w kraju. W V Klinice Chorób Wewnętrznych Śl. AM w Bytomiu opracowano system informacji o leku na komputer domowy ZX-Spectrum [12]. Wiele ośrodków podejmowało próbę przeniesienia na komputery IBM PC dzieła Podlewskich [10]; żadnej z tych prac jednak nie ukończono, a większości zaniechano [13, 14]. W bytomskiej placówce zrealizowano również program wspomagający leczenie nadciśnienia tętniczego [15]. Oparty o teorię zbiorów rozmytych algorytm optymalnego doboru leku zaproponował Tokarski [16]. W łódzkim Instytucie Medycyny Społecznej od dawna prowadzone są badania nad komputerowym wspomaganie leczenia nadciśnienia tętniczego i przewlekłej niewydolności krążenia [17, 18].

Zakres zagadnień uwzględnianych we wspomnianym systemie informacji był jednakże niewielki i obejmował indeks środków farmakologicznych, wskazania, przeciwwskazania oraz odnośniki do krajowych wydawnictw informacyjnych [10, 19-21]. Autorowi nie są znane próby budowy systemu informacji o leku opartego o szerszy przegląd piśmiennictwa. Również wymienione próby wspomaganie terapii chorób układu krążenia uwzględniają stosunkowo skromny zestaw leków.

W pracach tych w małym stopniu wykorzystywano nowoczesne metody reprezentacji wiedzy. Koncepcja przedstawiona w publikacji Brodziaka i wsp. [22] wykazuje pewne związki z techniką ram (por. 2.3), lecz w nawiązującej doń pracy Braczkowskiego [15] dominuje podejście algorytmiczne. Omówiony tam system jest jednak stale doskonalony.

Z powyższego przeglądu dotychczasowych krajowych dokonań wynika potrzeba i celowość dalszych badań w wymienionych dziedzinach.

## 0.2 Cel pracy.

Analiza omówionych prób konstrukcji systemu informacji o leku wykazała, że w pracach tych zbyt mało uwagi poświęcono opracowaniu logicznego modelu informacji o leku i ogólnej koncepcji systemu, kładąc główny nacisk na fizyczną organizację danych, efektywność przetwarzania itp. Występowały też trudności w stworzeniu zespołu, w którym kompetencje farmakologiczne i informatyczne byłyby zrównoważone [13]. Z reguły zakładano uwzględnienie leków z wszystkich ważniejszych grup farmakologicznych, co utrudniało opracowanie informacji w ramach jednolitego schematu i ze względu na dużą liczbę leków wymagało zaangażowania wielu osób lub prowadziło do znacznych uproszczeń w opisie leku.

Przedstawiona praca stanowi część przedsięwzięcia mającego na celu znalezienie formuły takiego systemu informacji o wąskiej grupie leków, który mógłby zostać zrealizowany w niewielkim zespole, a równocześnie zawierał dane i dostarczał odpowiedzi na odpowiednio wysokim poziomie. System taki powinien umożliwiać sprawne uzyskanie informacji o wybranej grupie leków, a także służyć aktywną pomocą w indywidualnym doborze leku dla danego pacjenta.

Realizacja tego przedsięwzięcia przebiegała przy ścisłej współpracy z farmaceutą klinicznym. Przygotowanie systemu od strony farmakologicznej omawia praca Marii Wójcik-Jawień [23]. Celem niniejszej pracy jest:

- 1\* - opracowanie informatycznej koncepcji systemu i logicznego modelu informacji o leku.
- 2\* - weryfikacja słuszności przyjętych założeń przez praktyczną realizację eksperymentalnego systemu.

Koncepcję systemu opracowano na podstawie analizy istniejących systemów i wniosków z krajowych prób ich konstrukcji. Klasyczny system informacji biernie dostarcza poszukiwanych wiadomości dotyczących leku, jednostki chorobowej itp. Najczęściej jednak lekarz sięga do wydawnictw informacyjnych wtedy, gdy chce zdecydować, który lek zastosować u pacjenta. W takiej sytuacji korzyść z użytkowania tradycyjnego systemu informacji jest ograniczona, gdyż lekarz musi sam przestudiować ukazującą się na ekranie okazałą masę informacji. Proponowana koncepcja zakłada wzbogacenie klasycznego systemu informacji o system doradczy, którego zadaniem jest pomoc w dokonaniu optymalnego wyboru leku w indywidualnym przypadku. Charakterystyczna dla systemu doradczego zdolność objaśniania uzyskanego rozwiązania wspomaga również funkcję informacyjną, pozwalając wyeksponować te właściwości leku, które spowodowały jego wybór (lub odrzucenie). Komunikaty objaśniające mogą być niekiedy cenniejsze od suchego opisu wg tradycyjnego podziału na wskazania i przeciwwskazania.

Wychodząc z założenia, że farmakologia nie jest jakąś szczególnie nietypową dyscypliną wiedzy, należy przypuszczać, iż zastosowanie metodologii, która sprawdziła się w odniesieniu do wielu innych dziedzin i tu powinno dać pozytywne rezultaty. Niektóre krajowe publikacje zawierają odmienne sugestie: Muc i Brodziak [24] przedstawiają koncepcję systemu informacji o leku, w którym złożony model danych jest odbiciem złożonych metod klasyfikacji leków. Brodziak i wsp. [22] wyrażają opinię, że wiedza dotycząca postępowania leczniczego (w tym farmakoterapii) z trudem daje się przedstawić przy użyciu reguł postaci JEŻELI - TO.

Niniejsza praca podejmuje polemikę z tymi poglądami poprzez konstrukcję standardowych narzędzi informatycznych, a następnie pokazanie, jak można ich użyć do budowy systemu informacji o leku na przykładzie leków krążeniowych. Postanowiono równolegle zastosować dwie technologie informatyczne: relacyjne bazy danych i systemy doradcze z regułową bazą wiedzy. Obydwie cechuje prostota i wiele udanych zastosowań, zatem ich wybór do konstrukcji eksperymentalnego systemu powinien okazać się korzystny. Analiza działania tego systemu pozwoli m.in. zbadać, czy standardowe algorytmy rozumowania przybliżonego (por. 2.6.2) stanowią dobre odwzorowanie postępowania lekarza dokonującego wyboru leku, a zwłaszcza, czy zapewniają przestrzeganie zasady "*primum non nocere*".

### 0.3 Realizacja.

Realizacja wyznaczonego celu przebiegała następująco:

- Sformułowano ogólne założenia tworzonego systemu.
- Opracowano schemat logiczny informacyjnej bazy danych.
- Zaprojektowano, napisano i uruchomiono program umożliwiający tworzenie i eksploatację relacyjnej bazy danych. Wykorzystując ten program zdefiniowano strukturę bazy danych przeznaczonej do przechowywania informacji o leku, a następnie wprowadzono informacje o 91 lekach stosowanych w leczeniu zaburzeń rytmu serca, dusznicy bolesnej i nadciśnienia tętniczego.
- Zaprojektowano, napisano i dołączono do systemu procedury, które wykorzystując istniejący już system zarządzania bazą danych (SZBD), pozwalają na tworzenie i użytkowanie systemu doradczego z wiedzą wyrażoną w postaci reguł wnioskowania. Przy pomocy tak wzbogaconego programu, współdziałając z farmaceutą klinicznym zgromadzono i sformalizowano wiedzę potrzebną do właściwego doboru leku.
- Przeanalizowano funkcjonowanie systemu.

#### 0.4 Podział tekstu.

Autor jest programistą, a praca ma charakter informatyczny. Jest jednak realizowana w Akademii Medycznej w ramach badań mających na celu usprawnienie informacji o leku. Uznano zatem za konieczne poprzedzenie rozważań merytorycznych dwoma rozdziałami wprowadzającymi w terminologię i problematykę baz danych oraz systemów doradczych. Pierwszy rozdział zawiera niezbędne definicje dotyczące relacyjnych baz danych i podaje ich podstawowe własności. Zagadnienie normalizacji ilustrowane jest przykładami dotyczącymi bazy danych o leku. Wprowadzone zostaje, zdefiniowane na użytek pracy, pojęcie symetrycznych składników klucza. W rozdziale drugim omawiane są podstawowe pojęcia dotyczące systemów ekspertowych i reprezentacji wiedzy. Zdefiniowane zostają pojęcia wnioskowania przybliżonego i zbiorów rozmytych. Trzeci rozdział przedstawia i analizuje założenia projektowe. Począwszy od ogólnych postulatów dotyczących całego systemu formułowane są założenia szczegółowe: projekt SZBD, projekt logicznej i fizycznej organizacji bazy danych o leku, projekt programu, który ma umożliwić tworzenie i eksploatację systemu doradczego; założenia bazy wiedzy tego systemu. Czwarty rozdział omawia szczegóły realizacji programu. Podane jest uzasadnienie wyboru języka programowania, przedstawione są stosowane struktury danych i podstawowe elementy programu. Rozdział piąty ilustruje wykorzystanie programu do skonstruowania systemu informacji i systemu doradczego dla leków krążeniowych. Ostatni, szósty rozdział zestawia możliwości programu i utworzonego przy jego pomocy systemu z przyjętymi wcześniej założeniami. Podane są przykłady wykorzystania systemu i końcowe wnioski.

# 1. BAZY DANYCH – WPROWADZENIE

## 1.1. Podstawowe pojęcia.

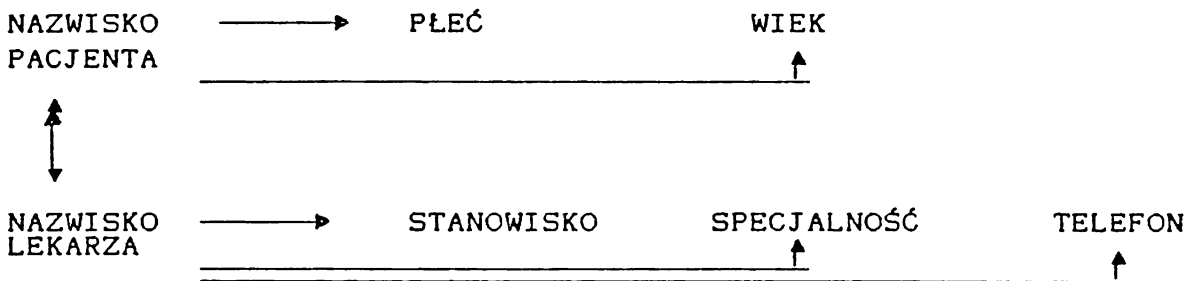
*Bazę danych (BD)* można zdefiniować jako zbiór wzajemnie powiązanych danych, mogących ulegać zmianom w czasie, pamiętanych bez zbędnej nadmiarowości (inaczej redundancji'), służących jednemu lub wielu zastosowaniom w sposób optymalny [2]. Dane przechowywane w BD mogą dotyczyć jednego lub kilku rodzajów *obiektów*. Opis obiektu składa się z *atrybutów*. Przykładowo, opis obiektu "PACJENT" może składać się z takich atrybutów, jak: NAZWISKO, PŁEĆ, WIEK, LEKARZ. Zapis dokonany w BD, a dotyczący obiektu nosi nazwę *rekordu*, zaś zapisy poszczególnych atrybutów nazywa się *polami* rekordu. W rozważanym przykładzie pola konkretnego rekordu mogą zawierać następujące dane:

NAZWISKO : Kowalski  
PŁEĆ : mężczyzna  
WIEK : 62  
LEKARZ : Dr Nowak

Zbiór możliwych wartości danego atrybutu nazywa się *dziedziną*. Np. dziedziną atrybutu PŁEĆ jest dwuelementowy zbiór {KOBIETA , MĘŻCZY-ZNA}, zaś dziedziną atrybutu WIEK - zbiór liczb nieujemnych. Informacje zawarte w bazie danych nie muszą ograniczać się do zawartości pól poszczególnych rekordów. Pomiędzy danymi mogą bowiem występować różnego rodzaju *powiązania*. Np., jeśli obok obiektów typu "PACJENT" w bazie danych występują obiekty typu "PERSONEL", o atrybutach NAZWISKO, STANOWISKO, SPECJALNOŚĆ, TELEFON, to pole LEKARZ w rozważanym przykładzie tworzy powiązanie pomiędzy obiektami obu wymienionych typów (tj "PACJENT" i "PERSONEL"). Istnienie tych powiązań pozwala na uzyskanie dodatkowych informacji, np. można się dowiedzieć, jaki jest telefon do lekarza, który leczy Kowalskiego, albo jaki jest średni wiek pacjentów Dr Wiśniewskiej. Wykaz obiek-



tów, ich atrybutów i istniejących powiązań tworzy *logiczny obraz danych*. Ta abstrakcyjna struktura, stanowiąca model wybranego wy-cinka realnego świata, często przedstawiana w postaci grafu (ryc. 1.1), nazywana bywa *schematem konceptualnym BD*. Natomiast rodzaj wykorzystywanych nośników informacji, rozmieszczenie danych w pa-mięci systemu komputerowego, organizacja dostępu do tych danych, stosowane techniki kompresji<sup>2</sup> itp. to czynniki określające *fizyczną reprezentację danych*.



Ryc. 1.1. Przedstawienie schematu BD w postaci grafu.

Aby dane przechowywane w BD można było wyszukiwać, aktualizować itp. konieczne są odpowiednie programy. Składają się one na *system zarządzania bazą danych (SZBD)*. Baza danych wraz z SZBD tworzy tzw. *system bazy danych*. W eksploatacji bazy danych ważną rolę odgrywa *administrator bazy danych*, tj. człowiek odpowiedzialny za zawartość BD, bezpieczeństwo i poufność danych itp.

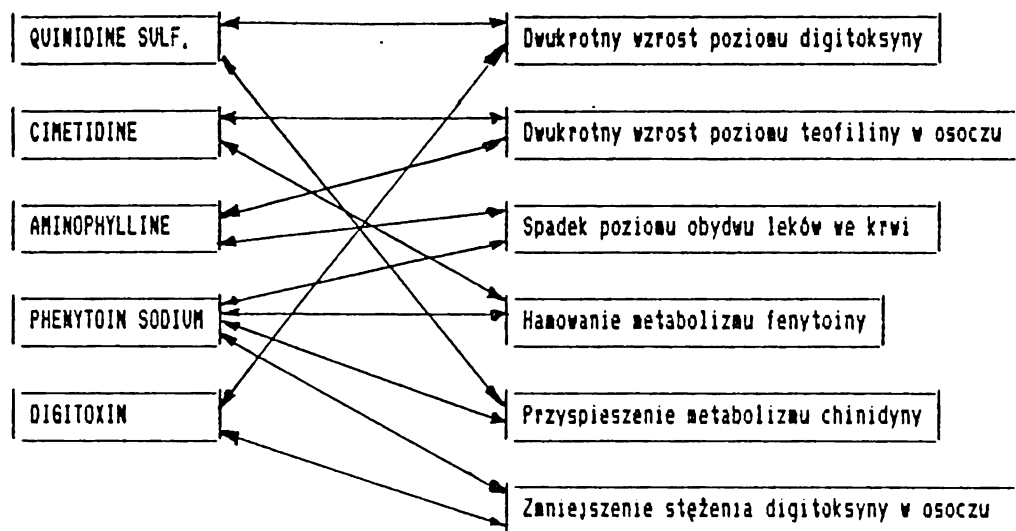
Podana wyżej definicja BD nie wyczerpuje jednak listy wymagań stawianym bazom danych i programom zarządzającym. Pełny wykaz tych wymagań zawiera cytowana już praca Martina [2]. Najważniejszymi postulatami wydają się być:

- *Postulat logicznej niezależności danych*. Wymaga on, aby związana z rozwojem bazy danych ewolucja schematu, wprowadzająca do BD nowe obiekty lub nowe atrybuty nie wymagała zmian w istniejących już programach korzystających z bazy danych.
- *Postulat fizycznej niezależności danych*. W myśl tego postulatu

fizyczny sposób przechowywania danych nie jest zależny ani od logicznej struktury danych, ani od użytkownika bazy danych (przez użytkownika rozumie się tutaj zarówno człowieka, jak i programy użytkowe odwołujące się do BD). Niezależność jest tu obopólna: organizacja fizyczna również nie może wpływać na strukturę logiczną, czy też sposób użytkowania bazy danych.

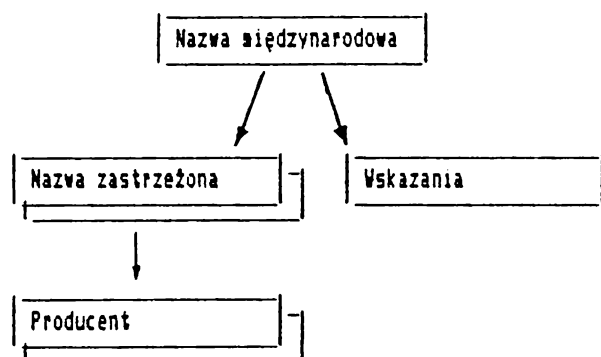
## 1.2. Modele danych.

Powiązania pomiędzy danymi dotyczącymi konkretnego zastosowania mogą być przedstawiane przy użyciu różnych modeli. W modelu sieciowym powiązania reprezentuje się w postaci sieci wskaźników łączących poszczególne dane elementarne (ryc. 1.2).



Ryc. 1.2. Baza danych o interakcjach leków przedstawiona w modelu sieciowym.

Model hierarchiczny podobnie określa powiązania pomiędzy danymi elementarnymi lub całymi obiektami, jednakże powiązania te są teraz jednokierunkowe (ryc. 1.3), tj. od zwierzchnika do podwładnego (inaczej: od poprzednika do następnika). W modelu tym nie można przedstawić niektórych powiązań obecnych w modelu sieciowym, jednakże hierarchiczne przedstawienie danych jest bardziej przejrzyste i odpowiada wielu zastosowaniom.



Ryc. 1.3. Przedstawienie danych w modelu hierarchicznym.

Model relacyjny polega na zestawieniu danych w formie tabelarycznej; powiązań pomiędzy danymi nie przedstawia się w nim w sposób jawny, gdyż wynikają one bezpośrednio ze struktury relacji (tabel). Okazuje się, że wszelkie powiązania, które można przedstawić w modelu sieciowym, dają się również przedstawić w postaci systemu relacyjnego [1], przy czym prezentacja relacyjna dla złożonych schematów jest powszechnie uważana za bardziej czytelną. W pracy przyjęto model relacyjny, w związku z tym wymaga on dokładniejszego przedstawienia.

### 1.3. Relacyjny model danych.

Zainteresowanie relacyjnym modelem danych zawdzięczamy pracom Codda [3, 4], które zapoczątkowały formalizację tego modelu. Publikacje te podały precyzyjne definicje szeregu pojęć związanych z relacyjnym podejściem do danych, przede wszystkim jednak zwróciły uwagę na rolę semantycznych związków pomiędzy danymi (zwłaszcza tzw. zależności funkcjonalnych) i korzyści płynące z postępowania zwanego normalizacją. Podstawowe fakty z tej dziedziny, przedstawione we współczesnym ujęciu, zostaną tu skrótowo (i nieformalnie) zreferowane; wyczerpujący wykład podejścia relacyjnego można znaleźć w licznych podręcznikach [1, 2, 25, 26].

W modelu relacyjnym dane przedstawiamy w postaci tabel, takich jak przykładowo zamieszczona na ryc. 1.4. Każdy wiersz tabeli odpowiada konkretnemu wystąpieniu rekordu w BD; w modelu relacyjnym wiersz taki nazywa się *krotką* (odpowiednik ang. *tuple*, termin polski wprowadzili tłumacze książki [1]). Pojedynczą tabelę nazywamy *relacją*. Odpowiada to dość ściśle formalnej (matematycznej) definicji relacji<sup>3</sup>. W myśl tej definicji:

- 1°. W relacji nie ma dwóch identycznych krotek.
- 2°. Porządek krotek relacji jest nieistotny, tj. zmiana tego porządku nie zmienia relacji.

W teorii relacyjnych baz danych dołącza się jeszcze trzeci warunek:

- 3°. Zamiana kolumn tabeli również nie zmienia informacji zawartej w tabeli i dlatego uważana jest za nieistotną.

W tym punkcie teoria ta rozmija się z matematycznym rozumieniem relacji.

Jeżeli tabela ma płaską strukturę - tj. wszystkie wiersze i kolumny ciągną się przez całą tabelę oraz wszystkie pola zawierają dane *atomowe* (niepodzielne, nie posiadające bardziej złożonej struktury) - to mówimy, że relacja jest w *pierwszej postaci normalnej* (1PN). Relacja z ryc. 1.4 jest w 1PN, natomiast przedstawiona na

ryc. 1.5 nie jest. Relację, która jest w IPN nazywamy *znormalizowana*.

Nazwa międzynarodowa	Naz. międzynar. w jęz. polskim	Nazwa chemiczna	Wzór sumaryczny	Mechanizm działania	Wskazania
AMIODARONE HCl	Amiodaron	chlorowodorek (2-butylo-3-benzofuranylo)-(4-[2-(dietylamino)-etoksy]-3,5-dijodofenilo) metanonu	$C_{26}H_{29}J_2NO_3 \cdot HCl$	Grupa III wg Williamsa, zaznaczone działanie $\alpha$ - i $\beta$ -adrenolityczne, bezpośredni wpływ na mięśniówkę naczyń	zaburzenia rytmu serca, także w zawale, choroba wieńcowa
FUROSENIDE	Furoseid	kwas 4-chloro-N-furfurylo-5-sulfamoiio-antranilowy	$C_{12}H_{11}ClN_2O_5S$	Diuretyk "pętlowy"	Nadciśnienie tętnicze, obrzęki, hiperkalcemia, stosowany celem wymuszenia diurezy w zatruciach
NITROGLYCERINE	Nitrogliceryna	trójazotan 1,2,3-propantriolu	$C_3H_6N_3O_9$	Działanie bezpośrednie na mięśniówkę naczyń	dusznicza bolesna, świeży zawał, przewł. niewydolność krążenia, obrzek płuc, zespół małego rzutu, dychawica oskrzelowa, kolka nerkowa, zatrucie cyjankami,

Ryc. 1.4. Relacja znormalizowana (przykład).

Nazwa międzynarodowa	Nazwa zastrzeżona	Producent	Kraj
AMIODARONE HCl	CORDAREX	LABAZ	D
	CORDARONE	LABAZ	F
		KRKA	YU
	CORBIONAX	ROLAND-MARIE	F
NITROGLYCERINE	NITROCARD	POLFA	PL
	NITROGLYCERINUM	UNIA	PL
		CASCAN	D
	NITROGLYCERIN	STADAPHARM	D
		RATIOPHARM	D
		WANDER	CH
STREULI		CH	
KABI	S		

Ryc. 1.5 Relacja nieznormalizowana.

### 1.3.1. Klucze relacji.

W relacji z ryc. 1.4 istnieje kilka atrybutów posiadających własność jednoznacznej identyfikacji krotek. Są nimi: 'Nazwa międzynarodowa', 'Nazwa chemiczna' i 'Nazwa międzynarodowa w języku polskim'. Atrybut: 'Wzór sumaryczny' takiej własności nie posiada, gdyż kilka zupełnie różnych substancji chemicznych może mieć identyczne wzory sumaryczne. Atrybut (lub zestaw atrybutów) jednoznacznie identyfikujący cały wiersz tabeli nazywa się *kluczem*. Każda relacja ma co najmniej jeden klucz<sup>4</sup>; może ich mieć wiele i wtedy nazywamy je *kluczmi kandydującymi*. Administrator BD decyduje, który klucz będzie używany do wyszukiwania krotek; taki klucz nazywamy *kluczem głównym*.

### 1.3.2. Schemat logiczny.

Gdy relacja jest znormalizowana, narysowanie nagłówka tabeli i wskazanie, które atrybuty tworzą klucz główny stanowi wygodną ilustrację schematu logicznego. Taki sposób przedstawiania schematu będzie stosowany w dalszej części pracy. Atrybuty wchodzące w skład klucza są wyróżniane przez otoczenie ich podwójną ramką. Na przykład ryc. 1.6 przedstawia schemat relacji z ryc. 1.4.

Nazwa międzynarodowa	Naz. międzynar. w jęz. polskim	Nazwa chemiczna	Wzór sumaryczny	Mechanizm działania	Wskazania
----------------------	--------------------------------	-----------------	-----------------	---------------------	-----------

Ryc. 1.6. Schemat logiczny relacji z ryc. 1.4.

### 1.3.3. Postaci normalne relacji.

Przedstawienie bardziej złożonych informacji w postaci pojedynczej relacji napotyka jednak na pewne trudności, polegające najczęściej na nadmiernej redundancji, czyli niepotrzebnym, wielokrotnym powtarzaniu tych samych informacji. W teorii baz danych wyróżnia się obecnie pięć postaci normalnych relacji. Sprowadzenie zbioru relacji do wyższej postaci normalnej związane jest zawsze ze zmniejszeniem redundancji.

W relacji przedstawionej na ryc. 1.7 klucz główny składa się z dwóch atrybutów: 'Nazwa międzynarodowa' oraz 'Postać leku'. Dopiero podanie wartości obu tych atrybutów jednoznacznie wskazuje wiersz tabeli. Widać jednak, że wiele danych w owej tabeli niepotrzebnie się powtarza; wynika to stąd, że niektóre atrybuty są jednoznacznie wyznaczone już przez nazwę leku, a więc przez część (podzbiór właściwy) klucza. Przedstawienie tych samych danych w postaci dwóch tabel (relacji) likwiduje ten problem (ryc. 1.8). O relacjach, w których ów problem nie występuje mówimy, że są w *drugiej postaci normalnej (2PN)*. Ściślej: Relacja jest w 2PN, jeśli jest w 1PN i nie ma w niej atrybutów, które byłyby jednoznacznie wyznaczone już na podstawie znajomości części (a nie całego) klucza. Wiadomo, że dowolną relację będącą w 1PN da się zastąpić reprezentującym te same informacje zbiorem relacji, z których każda jest w 2PN [4].

Nazwa międzynarodowa	Nazwa chemiczna	Wzór sumaryczny	Postać leku	Dawkowanie
NITROGLYCERINE	trójazotan 1,2,3-propantriolu	$C_3H_5N_3O_9$	tabletki	0,5-1 mg podjęzykowo
NITROGLYCERINE	trójazotan 1,2,3-propantriolu	$C_3H_5N_3O_9$	krople	3-6 kropli doustnie
NITROGLYCERINE	trójazotan 1,2,3-propantriolu	$C_3H_5N_3O_9$	kapsułki retard	1 kapsułka co 12 h
NITROGLYCERINE	trójazotan 1,2,3-propantriolu	$C_3H_5N_3O_9$	maść	wcierać 3dziennie
DIPROPHYLLINE	7-(2,3-dihydroksypropylo)-teofilina	$C_{10}H_{14}N_4O_4$	tabletki	200 mg 3dziennie
DIPROPHYLLINE	7-(2,3-dihydroksypropylo)-teofilina	$C_{10}H_{14}N_4O_4$	czopki	400 mg 3dziennie

Ryc. 1.7. Przykład relacji, w której nie wszystkie atrybuty zależą od całego klucza.

Nazwa międzynarodowa	Nazwa chemiczna	Wzór sumaryczny
NITROGLYCERINE	trójazotan 1,2,3-propantriolu	$C_3H_5N_3O_9$
DIPROPHYLLINE	7-(2,3-dihydroksypropylo)-teofilina	$C_{10}H_{14}N_4O_4$

Nazwa międzynarodowa	Postać leku	Dawkowanie
NITROGLYCERINE	tabletki	0,5-1 mg podjęzykowo
NITROGLYCERINE	krople	3-6 kropli doustnie
NITROGLYCERINE	kapsułki retard	1 kapsułka co 12 h
NITROGLYCERINE	maść	wcierać 3dziennie
DIPROPHYLLINE	tabletki	200 mg 3dziennie
DIPROPHYLLINE	czopki	400 mg 3dziennie

Ryc. 1.8. Dwie relacje będące wynikiem sprowadzenia relacji z ryc. 1.7 do 2PN.

Relacja przedstawiona na ryc. 1.9 jest w 2PN, nie jest ona jednak wolna od wad. Również tu pewne informacje niepotrzebnie się powtarzają. Kluczem głównym tej relacji jest 'Nazwa zastrzeżona'.



Jednakże 'Podstawowe działanie' jest jednoznacznie wyznaczone przez informację w polu 'Grupa farmakologiczna'. Mankament ten usuwamy, podobnie jak poprzednio, zastępując wyjściową relację dwiema, tak jak to przedstawia ryc. 1.10. O relacjach tych mówimy, że są w trzeciej postaci normalnej (3PN).

Nazwa zastrzeżona	Wzór sumaryczny	Grupa farmakologiczna	Podstawowe działanie
FUROSEMIDE	$C_{12}H_{11}ClN_2O_5S$	Diureticum	moczopędne
DIPROPHYLLINE	$C_{10}H_{14}N_4O_4$	Diureticum	moczopędne
PRazosin HCl	$C_{19}H_{21}N_5O_4 \cdot HCl$	Antihypertonicum	obniżające ciśnienie krwi
MEXILETINE HCl	$C_{11}H_{17}NO \cdot HCl$	Antiarrhythmicum	przeciwarhythmiczne
AMIODARONE HCl	$C_{25}H_{29}J_2NO_3 \cdot HCl$	Antiarrhythmicum	przeciwarhythmiczne
CARBOCROMEN HCl	$C_{20}H_{27}NO_6 \cdot HCl$	Vasodilatans	rozszerzające naczynia
AJMALINE	$C_{20}H_{26}N_2O_2 \cdot C_2H_6O$	Antiarrhythmicum	przeciwarhythmiczne

Ryc. 1.9. Przykład relacji, w której występują przechodnie zależności funkcjonalne.

Nazwa zastrzeżona	Wzór sumaryczny	Grupa farmakologiczna
FUROSEMIDE	$C_{12}H_{11}ClN_2O_5S$	Diureticum
DIPROPHYLLINE	$C_{10}H_{14}N_4O_4$	Diureticum
PRazosin HCl	$C_{19}H_{21}N_5O_4 \cdot HCl$	Antihypertonicum
MEXILETINE HCl	$C_{11}H_{17}NO \cdot HCl$	Antiarrhythmicum
AMIODARONE HCl	$C_{25}H_{29}J_2NO_3 \cdot HCl$	Antiarrhythmicum
CARBOCROMEN HCl	$C_{20}H_{27}NO_6 \cdot HCl$	Vasodilatans
AJMALINE	$C_{20}H_{26}N_2O_2 \cdot C_2H_6O$	Antiarrhythmicum

Grupa farmakologiczna	Podstawowe działanie
Diureticum	moczopędne
Antihypertonicum	obniżające ciśnienie krwi
Antiarrhythmicum	przeciwarhythmiczne
Vasodilatans	rozszerzające naczynia

Ryc. 1.10. Dwie relacje powstałe w wyniku sprowadzenia do 3PN relacji z ryc. 1.9.

Ogólnie: Relacja jest w 3PN, jeśli jest w 2PN i nie ma w niej atrybutów nie wchodzących w skład klucza, ale wyznaczających jednoznacznie niektóre inne atrybuty. Jednoznaczne wyznaczanie atrybutu (zbioru atrybutów) przez inny atrybut (zbiór atrybutów) nazywa się *zależnością funkcjonalną*. Oczywiście wszystkie atrybuty relacji są

funkcjonalnie zależne od klucza; jeśli prócz tego występują przechodnie zależności funkcjonalne, to relacja nie jest w 3PN.

Również dowolną relację w 2PN można zastąpić zbiorem relacji będących w 3PN [4].

Ilość powtarzających się informacji w relacji z ryc. 1.11 jest znaczna, choć, jak łatwo się przekonać, spełnia ona warunki 3PN.

Nazwa międzynarodowa	Układ, na który jest wywierane dział. uboczne	Preparat dostępny w kraju
QUINIDINE SULF.	pokarmowy	CHINIDINUM SULFURICUM
QUINIDINE SULF.	pokarmowy	KINIDIN DURULES
QUINIDINE SULF.	nerwowy	CHINIDINUM SULFURICUM
QUINIDINE SULF.	nerwowy	KINIDIN DURULES
QUINIDINE SULF.	krażenia	CHINIDINUM SULFURICUM
QUINIDINE SULF.	krażenia	KINIDIN DURULES
NIFEDIPINE	pokarmowy	CORDAFEN
NIFEDIPINE	pokarmowy	CORINFAR
NIFEDIPINE	nerwowy	CORDAFEN
NIFEDIPINE	nerwowy	CORINFAR
NIFEDIPINE	krażenia	CORDAFEN

Ryc. 1.11 Przykład relacji wymagającej normalizacji do 4PN.

I znów zastąpienie powyższej relacji dwiema, przedstawionymi na ryc. 1.12 eliminuje nadmiarowość danych.

Nazwa międzynarodowa	Układ, na który jest wywierane dział. uboczne
QUINIDINE SULF.	pokarmowy
QUINIDINE SULF.	nerwowy
QUINIDINE SULF.	krażenia
NIFEDIPINE	pokarmowy
NIFEDIPINE	nerwowy
NIFEDIPINE	krażenia
NIFEDIPINE	krew

Nazwa międzynarodowa	Preparat dostępny w kraju
QUINIDINE SULF.	CHINIDINUM SULFURICUM
QUINIDINE SULF.	KINIDIN DURULES
NIFEDIPINE	CORDAFEN
NIFEDIPINE	CORINFAR

Ryc. 1.12 Wynik sprowadzenia relacji z ryc. 1.11 do 4PN

Ten rodzaj nadmiarowości wiąże się z występowaniem w relacji tzw. *zależności wielowartościowych*. Zależność wielowartościowa polega na wyznaczaniu z b i o r u wartości pewnego atrybutu przez inny atrybut. W rozważanym przykładzie występują dwie zależności wielowartościowe:

1. nazwa leku określa zbiór preparatów dostępnych w kraju
2. nazwa leku wyznacza zbiór układów, na które lek wywiera działanie uboczne.

Normalizacja do 4PN polega na takim rozbiciu wyjściowej relacji, aby każda z relacji wynikowych zawierała najwyżej jedną zależność wielowartościową.

We wszystkich omawianych dotychczas PN nadmiarowość danych widoczna była na pierwszy rzut oka. Przejście do 5PN wymaga dostrzeżenia bardziej subtelnych przypadków nadmiarowości.

Pacjent	Lek (nazwa międzynarodowa)	Producent
Kowalski	NIFEDIPINE	BAYER
Kowalski	ACETYLSALICYLIC ACID	BAYER
Malinowski	NITROGLYCERINE	POLFA
Malinowski	NITROGLYCERINE	MERCK
Malinowski	DIGOXIN	POLFA
Malinowski	DIGOXIN	SANDOZ
Malinowski	DIGOXIN	BOEHRINGER MANNHEIM
Wiśniewski	FUROSEMIDE	HOECHST

Ryc. 1.13 Relacja wymagająca normalizacji do 5PN

Tabela przedstawiona na ryc. 1.13 zawiera wykaz leków, które wymienieni pacjenci są skłonni zażywać. Zakładamy, przy tym, że pacjent zażywa tylko leki zalecone przez lekarza, ale preparaty dobiera wg własnego zaufania do producenta. Nadmiarowość w tej relacji ujawnia się np. w ten sposób, że informacja o tym, że Kowalski ma zaufanie do firmy «Bayer» powtarza się kilkakrotnie. Również kilka razy podana jest informacja, że Malinowski powinien zażywać digo-

ksynę i nitroglicerynę. Nadmiarowości tej można uniknąć zastępując wyjściową relację trzema przedstawionymi na ryc. 1.14, wszystkie bowiem informacje z tabeli z ryc. 1.13 dadzą się z nich wywnioskować.

Pacjent	Lek (nazwa międzynarodowa)
Kowalski	NIFEDIPINE
Kowalski	ACETYLSALICYLIC ACID
Malinowski	NITROGLYCERINE
Malinowski	DIGOXIN
Wiśniewski	FUROSEMIDE

Pacjent	Producent
Kowalski	BAYER
Malinowski	POLFA
Malinowski	MERCK
Malinowski	SANDOZ
Malinowski	BOEHRINGER MANNHEIM
Wisniewski	HOECHST

Lek (nazwa międzynarodowa)	Producent
NIFEDIPINE	BAYER
ACETYLSALICYLIC ACID	BAYER
NITROGLYCERINE	POLFA
NITROGLYCERINE	MERCK
DIGOXIN	POLFA
DIGOXIN	SANDOZ
DIGOXIN	BOEHRINGER MANNHEIM
FUROSEMIDE	HOECHST

Ryc. 1.14. Trzy relacje w SPN równoważne relacji z ryc. 1.13

Przedstawiony proces stopniowej normalizacji pozwala niejednokrotnie na znaczne zmniejszenie szkodliwej redundancji, nie gwarantuje jednak całkowitego jej wyeliminowania. Relacja spełniająca wymogi wszystkich omówionych postaci normalnych może wykazywać wy-

soka redundancje z powodu istnienia w niej symetrii względem pewnych atrybutów. Analizę tego problemu i proponowane środki zaradcze przedstawiono w punkcie 1.3.4.

#### 1.3.4. Symetryczne składniki klucza.

W czasie projektowania relacji INTERAKCJE bazy danych o lekach krążeniowych (por. 3.5) powstała potrzeba określenia nowego<sup>5</sup> pojęcia *symetrycznych atrybutów*.

Definicja: Atrybuty  $A_1, A_2, \dots, A_k$  mające wspólną dziedzinę ( $k \leq n$ ,  $n$  - ilość atrybutów relacji) nazywamy symetrycznymi, wtedy i tylko wtedy, gdy z faktu występowania w relacji krotki

$(a_1, a_2, \dots, a_k, a_{k+1}, \dots, a_n)$

wynika występowanie w tej samej relacji krotki

$(b_1, b_2, \dots, b_k, a_{k+1}, \dots, a_n)$

gdzie  $(b_1, b_2, \dots, b_k)$  jest dowolną permutacją wartości

$(a_1, a_2, \dots, a_k)$  atrybutów  $A_1, A_2, \dots, A_k$ .

Np. relacja zawierająca odległości pomiędzy miejscowościami może mieć schemat przedstawiony na ryc. 1.15.

Miejscowość 1	Miejscowość 2	Odległość
---------------	---------------	-----------

Ryc. 1.15. Relacja z symetrycznymi atrybutami.

Atrybuty 'Miejscowość 1' i 'Miejscowość 2' są w tej relacji symetryczne: jeśli krotka (Kraków, Zakopane, 106) występuje w relacji to musi w niej być też krotka (Zakopane, Kraków, 106).

Przechowywanie obydwu tych zapisów prowadzi do znacznej redundancji - każda informacja zapisywana jest dwukrotnie (przy  $k$  symetrycznych atrybutach  $k!$ -krotnie!). Redundancji tej nie da się usunąć na drodze rozbicia na inne relacje, musi więc tego dokonać SZBD. Uwzględnienie atrybutów symetrycznych wymaga od projektanta systemu zarządzania bazą danych stworzenia administratorowi BD możliwości deklarowania, które atrybuty są symetryczne (na etapie określania struktury logicznej) i zapewnienie efektywnej obsługi tej relacji w czasie eksploatacji BD. W niniejszej pracy zastosowano następujące rozwiązanie: SZBD sortuje wartości wchodzących w skład klucza

symetrycznych atrybutów (stosownie do naturalnego uporządkowania ich wspólnej dziedziny) i taką krotkę zapisuje fizycznie w bazie danych. Podobnie postępuje się przy odczycie i innych operacjach. W rozwiązaniu tym celowo ograniczono się do atrybutów wchodzących w skład klucza, gdyż atrybuty symetryczne najczęściej pojawiają się właśnie w kluczu.

### 1.3.5. Operacje relacyjne.

Relacje mogą ulegać przekształceniom w wyniku wykonywania na nich różnych działań (operacji). Przykładem takich operacji są przekształcenia relacji BD z niższej do wyższej postaci normalnej.

Algebraiczna analiza przekształceń relacji pozwoliła wyróżnić niewielką ilość podstawowych działań relacyjnych [1,25-27]; inne operacje są wynikiem złożenia tych podstawowych działań. Do najważniejszych działań relacyjnych należą: *projekcja*, *selekcja* oraz *połączenie*. Projekcję i selekcję wykonuje się na jednej relacji, połączenie jest działaniem dwuargumentowym. Wynikiem każdego z tych działań jest pojedyncza relacja.

Projekcja polega na pozostawieniu w tabeli przedstawiającej relację tylko tych kolumn, które odpowiadają wybranym atrybutom oraz wykreśleniu pozostałych; aby tak utworzona tabela przedstawiała relację należy usunąć z niej powtarzające się wiersze.

Z kolei selekcja polega na pozostawieniu w relacji tych wierszy, które spełniają wskazany warunek i usunięciu pozostałych.

W ogólnym przypadku operacja połączenia może być dość złożona. Najczęściej mamy do czynienia z *połączeniem naturalnym*. W obydwu łączonych tabelach wyróżniamy po jednym atrybucie; atrybuty te powinny mieć jednakowe dziedziny. Tworzymy trzecią tabelę, zawierającą atrybuty pochodzące z obydwu tabel i wpisujemy w niej wszystkie wiersze jakie można utworzyć korzystając z danych zawartych w wyjściowych tabelach, przestrzegając zasady, aby wartości obu wy-

różnionych atrybutów były równe.

Wszystkie wymienione operacje mają duże znaczenie w procesie wyszukiwania potrzebnych danych. Selekcja pozwala wyszukać dane spełniające określony warunek, projekcja daje możliwość ograniczenia się tylko do interesujących atrybutów. Dzięki połączeniu można zrealizować powiązania istniejące pomiędzy relacjami bazy danych. Dla umożliwienia odpowiedzi na pytania użytkownika SZBD powinien być wyposażony w środki umożliwiające wykonanie wymienionych operacji. Zwraca na to uwagę Codd w jednej z późniejszych prac [28], wzbogacając definicję systemu relacyjnego o postulat wyposażenia takiego systemu w operacje relacyjne.

### 1.3.6. Języki zapytań do bazy danych.

SZBD musi umożliwiać użytkownikowi formułowanie zapytań do bazy danych (ang. *query*). W tym celu SZBD oferuje użytkownikowi tzw. język zapytań. Sformułowanie pytania w takim języku polega albo na określeniu warunków jakie mają spełniać poszukiwane krotki (języki oparte o rachunek predykatów [29]) lub też podaniu działań, które należy wykonać na relacjach bazy danych aby uzyskać odpowiedź (języki oparte o algebrę relacji Codd'a [27]). Języki te są szczególnie omawiane w cytowanym piśmiennictwie [1,25]. Język relacyjny posiada duże znaczenie teoretyczne; moc innych języków często jest określana przez porównanie do systemu relacyjnego. Jednakże sformułowanie pytania ma charakter proceduralny, tj. polega na zdefiniowaniu drogi poszukiwań (tzw. nawigacja programisty [2]). Język predykatów wydaje się bardziej naturalnym sposobem formułowania pytań.

Szczególnie interesującym językiem, zaprojektowanym z myślą o nieprofesjonalnych użytkownikach komputerów jest QUERY BY EXAMPLE opracowany w 1975 roku przez Zloofa [30]. Jego szczegółowy opis można znaleźć w klasycznych podręcznikach baz danych [1,25]. Język



ten łączy w sobie dużą moc z niezwykle prostotą użytkowania. Jest językiem całkowicie nieproceduralnym i w dobie powszechnego tworzenia "przyjaznych" programów (ang. *user friendly*) może służyć za wzór godny naśladowania. W niniejszej pracy zaproponowano znacznie skromniejszy język, jego projekt jest jednak w znacznej mierze inspirowany ideami zawartymi w QUERY BY EXAMPLE.

## 2. WPROWADZENIE DO SYSTEMÓW DORADCZYCH

### 2.1. Systemy doradcze jako kierunek badań nad sztuczną inteligencją.

Dziedzina sztucznej inteligencji (SI) stara się tworzyć programy i systemy komputerowe, które w pewnym zakresie funkcjonują tak, iż sprawiają wrażenie inteligentnego działania [5]. Np. programy komputerowe grające (i nierzadko wygrywające!) w szachy stanowią niewątpliwym przykład sukcesów tej dziedziny informatyki. SI od dawna zajmuje się zagadnieniem rozwiązywania różnych problemów na drodze automatycznego wnioskowania. Początkowo wierzono, że określając odpowiednie prawa wnioskowania, przy pomocy komputerów o wciąż rosnącej mocy obliczeniowej, będzie można rozwiązywać dowolnie złożone, ogólne problemy. Niepowodzenie przedsięwzięcia opartego o te założenia (General Problem Solver, GPS [31]), skierowało wysiłki badaczy w kierunku, który okazał się niezwykle owocny. Podjęto mianowicie próby uzyskania zadowalających rezultatów przy rozwiązywaniu problemów dotyczących wąskich, specjalistycznych dziedzin. Realizując kolejne projekty [32,33] zrozumiano, że na sprawność systemu mniejszy wpływ ma wybór algorytmu wnioskowania, natomiast decyduje o niej zgromadzona w tym systemie wiedza. Według Feigenbauma [6] wiedza jest kluczem do sprawności, natomiast reprezentacja wiedzy oraz strategię wnioskowania są mechanizmami umożliwiającymi jej wyzyskanie.

Uwzględniając profil niniejszej pracy, na szczególną uwagę wśród pierwszych systemów realizowanych wg powyższej zasady zasługuje MYCIN [32], program komputerowy diagnozujący infekcje bakteryjne i proponujący, stosownie do rozpoznania odpowiedni lek. Wiele rozwiązań zawartych w tym systemie wykorzystano w innych, głośniejszych pracach [33,34,35]. Systemy tego rodzaju, ze względu na wysoką jakość dostarczanych przez nie rozwiązań, przyjęto nazywać *systemami*

*ekspertowymi* (por. przypis 1 we wstępie).

Systemy doradcze dobitnie dowiodły swej przydatności w praktyce. Oto dwa przykłady: PROSPECTOR [34], system wspomagający poszukiwania geologiczne przyczynił się do odkrycia pokaźnego złoża molibdenu, wartości około 100 mln. dolarów. CADUCEUS [36] jest systemem medycznym, który dysponuje przewyższającą możliwości człowieka wiedzą z zakresu medycyny wewnętrznej (ocenia się, że wiedza systemu stanowi około 85% całej wiedzy istniejącej w tym zakresie); podczas testów system prawidłowo diagnozował przypadki, z którymi lekarze-eksperci nie mogli sobie poradzić.

## 2.2. Podstawowe definicje.

*System doradczy (ekspertowy) (SD)* to program komputerowy, który ze sprawnością dorównującą lub przewyższającą człowieka-specjalistę (eksperta) potrafi rozwiązywać problemy z zakresu pewnej, zazwyczaj wąskiej dyscypliny wiedzy. Powyższa próba definicji SD jest zdecydowanie niepełna; dwie dekady rozwoju systemów ekspertowych nie przyniosły co prawda formalnej definicji SD, określiły jednak szereg cech dlań charakterystycznych. *Idealny system doradczy* [6]: wyposażony jest w reguły postępowania, którymi kieruje się w rozwiązywaniu problemów w swej dziedzinie, unika bezowocnych poszukiwań, dostarcza rozwiązań wysokiej jakości, wnioskowanie przeprowadza na drodze manipulowania symbolami, zna podstawowe prawa rządzące jego dziedziną, posiada możliwość rozumowania w warunkach wiedzy niepełnej i niepewnej. Potrafi rozwiązywać trudne problemy ze swej złożonej dziedziny; posiada umiejętność wyjaśniania rozumowania, które doprowadziło do rozwiązania. Problem wyrażony przez użytkownika w języku naturalnym przekształca do własnej, wewnętrznej reprezentacji, umożliwiając mu przeprowadzenie rozumowania. Ma zdolność rozumowania na temat swej wiedzy (lub jej braku), zwłaszcza w celu odtworzenia własnej drogi rozwiązywania problemu dla wy-

jaśnienia wniosków.

Posługiwanie się informacją symboliczną jest cechą charakterystyczną nie tylko dla SD, lecz także dla innych programów sztucznej inteligencji. W SD informację symboliczną stanowi wiedza systemu. Twórca systemu musi znaleźć odpowiedni sposób przechowywania tej wiedzy w pamięci komputera, czyli wybrać właściwą *reprezentację wiedzy*.

### 2.3. Reprezentacja wiedzy.

Do najczęściej stosowanych środków przedstawiania wiedzy należą:

- algorytmy
- język logiki
- reguły wnioskowania
- ramy

Klasyczne programy komputerowe zawierają często znaczny zasób wiedzy, dzięki której są w stanie rozwiązywać złożone zadania. Np. programy stosowane w terapii monitorowanej stężeniem leku [37, 38, 39] skupiają w sobie wiedzę dotyczącą modeli farmakokinetycznych, statystyki, technik rozwiązywania złożonych równań różniczkowych i optymalizacji (tj. metod znajdowania minimum funkcji wielu zmiennych). Wiedza taka zostaje wcielona do programu w postaci ciągu poleceń dla komputera, wyrażonych w wybranym języku programowania, zgodnie z wcześniej opracowanym przepisem, czyli algorytmem. Mówimy, że wiedza jest przedstawiona w sposób *proceduralny*. Taka reprezentacja wiedzy szczególnie dobrze nadaje się do rozwiązywania problemów numerycznych, kombinatorycznych itp., dla których znane są efektywne algorytmy.

Całkowicie odmienny charakter mają metody reprezentacji wiedzy oparte o logikę. Polegają one na wyrażaniu wiedzy o danej dziedzinie w sposób *deklaratywny*. W przeciwieństwie do proceduralnego zapisu wiedzy określa się *c o w i a d o m o* o poszukiwanym rozwią-

zaniu, a nie jak to rozwiązanie uzyskać. Używany przy tym formalizm wywodzi się z rachunku predykatów pierwszego rzędu [40]. Najpopularniejszym systemem programowania w języku logiki jest PROLOG [41,42]. Stosowanie logiki jest szczególnie celowe w sytuacjach, gdy wiedza ma charakter ścisłych relacji pomiędzy obiektami i daje się łatwo wyrazić w postaci zbioru postulatów i związków przyczynowych.

Reguły wnioskowania wyrażają pewne ogólne prawa rządzące dziedziną SD. Zapisywane są one jako zdania postaci JEŻELI... TO..., np. JEŻELI pacjent jest w podeszłym wieku TO  $\beta$ -blokey nie są dla niego wskazanymi lekami. Również w tej technice przeważa deklaratywny charakter przedstawienia wiedzy. Reprezentacja regułowa może się okazać korzystniejsza od logiki w dziedzinach empirycznych, w których dominują rozwiązania heurystyczne. Systemy regułowe dobrze się nadają do wnioskowania w warunkach niepewnej wiedzy [43].

Opis obiektów w postaci ram (ang. *frames*) sprowadza się do wypełnienia odpowiednich formularzy, takich jak przedstawione na ryc. 4.1. Elementy takiego formularza nazywane są *klatkami*, wewnątrz klatek wyróżnia się tzw. *fasety* [7,44,45]. Ramy umożliwiają przedstawienie hierarchii obiektów. Do tego celu wykorzystywane są specjalne klatki: KLASA NADRZĘDNA oraz EGZEMPLARZE. Klatka EGZEMPLARZE występuje w ramach opisujących klasy obiektów. Ramy takie stanowią prototyp (wzorzec) dla ram-egzemplarzy dotyczących konkretnych obiektów lub podklas (konkretny obiekt można traktować jak klasę najniższego poziomu). Pozostałe klatki opisują własności obiektu lub klasy, której rama dotyczy. Odpowiadają one pojęciu atrybutu w bazie danych. Fasety związane z klatką służą m.in. do przechowywania wartości atrybutu, typu jego wartości itp. Przy tworzeniu ramy-egzemplarza dziedziczy ona cechy klasy wskazanej przez KLASA NADRZĘDNA. Fasety GDY POTRZEBNE i GDY UMIESZCZONE zawierają procedury, które mają zostać wykonane, odpowiednio wtedy,

gdy wartość klatki jest pobierana lub wpisywana. Tego rodzaju procedury nazywane są *demonami*. Typowym zadaniem demona jest sprawdzenie, czy wpisywana do klatki wartość jest poprawna. Również niektóre klatki mogą zawierać procedury, których zadaniem jest udzielanie odpowiedzi na stawiane ramie pytania. Te procedury nazywa się *siugami*.

JEDNOSTKA: diuretyki  
KLASA NADRZĘDNA: leki hipotensyjne  
EGZEMPLARZE: diuretyki tiazydowe, diuretyki pętlowe,  
diuretyki oszczędzające potas, diuretyki  
ksantynowe

KLATKA: działanie  
TYP WARTOŚCI: rodzaj działania  
WARTOŚĆ: moczopędne

JEDNOSTKA: diuretyki ksantynowe  
KLASA NADRZĘDNA: diuretyki  
EGZEMPLARZE: aminofilina, diprofilina, teofilina

KLATKA: mechanizm działania  
TYP WARTOŚCI: tekst  
WARTOŚĆ: zwiększenie filtracji kłębkowej

KLATKA: maksymalny efekt diuretyczny [%]  
TYP WARTOŚCI: liczba  
WARTOŚĆ: 1

JEDNOSTKA: aminofilina  
KLASA NADRZĘDNA: diuretyki ksantynowe

KLATKA: Objętość dystrybucji [l]  
TYP WARTOŚCI: liczba  
WARTOŚĆ: nieznana  
GDY POTRZEBNE: ( oszacowanie  $V_d$  wg równania  
 $V_d = 0.5 \times BW$  [46] )

KLATKA: Dawka [mg]  
TYP WARTOŚCI: liczba  
WARTOŚĆ: nieznana  
WARTOŚĆ DOMYŚLNA: 350  
GDY POTRZEBNE: ( procedura obliczania dawki na pod-  
stawie oznaczeń stężenia leku  
w krwi, np. metodą bayesowską  
[37,38] )  
GDY UMIESZCZONE: ( procedura sprawdzająca, czy nie  
został przekroczony zakres  
terapeutyczny)

Ryc. 2.1. Przykład reprezentacji wiedzy przy wykorzystaniu ram.

Technika ram pozwala na zespolenie w jednej reprezentacji rozwiązań algorytmicznych i deklaratywnych. Ramy stosuje się zwłaszcza w sy-

tuacjach, w których zachodzi potrzeba opisu złożonych obiektów o rozbudowanej hierarchii.

#### 2.4. Strategie wnioskowania.

Stosowanie deklaratywnego przedstawienia wiedzy na komputerach o architekturze zgodnej z koncepcją von Neumanna<sup>1</sup>, a więc preferującą reprezentację algorytmiczną, wymaga symulacji tzw. maszyny wnioskującej (ang. *inference engine*) w sposób programowy. W tym celu konieczne jest określenie *strategii wnioskowania*, czyli zasad postępowania, mających na celu wykorzystanie wiedzy systemu do znalezienia rozwiązania. Punktem wyjścia dla procesu wnioskowania mogą być zgromadzone dane albo cel, który należy osiągnąć. W pierwszym przypadku mówimy o *wnioskowaniu w przód*, czyli *sterowanym danymi* (ang. *forward chaining, data driven*). Drugi przypadek określany jest jako *wnioskowanie wstecz*, a więc *sterowane celem* (ang. *backward chaining, goal driven*). Zgromadzone dane, lub te, które mają zostać ustalone w czasie wnioskowania nazywać będziemy *stwierdzeniami*. *Stwierdzenia* mają charakter statyczny i opisują aktualny (bądź przeszły) stan rzeczy w formie zbliżonej do zdania języka naturalnego, np. PACJENT JEST W PODESZŁYM WIEKU (mamy tu na myśli konkretnego pacjenta). Wnioskowanie wstecz polega więc na wyszukiwaniu tych reguł, które w swych konkluzjach wypowiadają się na temat stwierdzenia będącego celem wnioskowania. Przesłanki tych reguł są kolejno analizowane. Przesłanka jest koniunkcją pewnych stwierdzeń. O ile ocena logiczna tych stwierdzeń nie jest jeszcze znana, należy wykonać wnioskowania, których celami będą te nowe stwierdzenia. Gdy ten rekurencyjny proces nie da rezultatu, "ostatnią deską ratunku" jest zapytanie użytkownika o prawdziwość rozważanego stwierdzenia. Jeśli dla którejś z rozważanych reguł uda się ustalić prawdziwość jej przesłanki, następuje "odpalenie" (ang. *fire*) reguły, w rezultacie czego zostaje ustalona prawdziwość stwierdzeń

## 2.5. Pozyskiwanie wiedzy.

Konstrukcja systemu doradczego wymaga przede wszystkim zgromadzenia i usystematyzowania wiedzy z rozważanej dziedziny, a następnie wyrażenia tej wiedzy w odpowiednio dobranej reprezentacji. Proces ten, zwany *pozyskiwaniem wiedzy* (ang. *knowledge acquisition*), wymaga współpracy specjalisty (eksperta) oraz tzw. *inżyniera wiedzy* (ang. *knowledge engineer*) - osoby dobrze znającej technologię systemów doradczych, posiadającej również pewną orientację w dziedzinie tworzonego systemu. Jednym z najważniejszych zadań technologii systemów doradczych jest dążenie do usprawnienia (m.in. środkami informatycznymi) procesu pozyskiwania wiedzy.

## 2.6. Rozumowanie przybliżone.

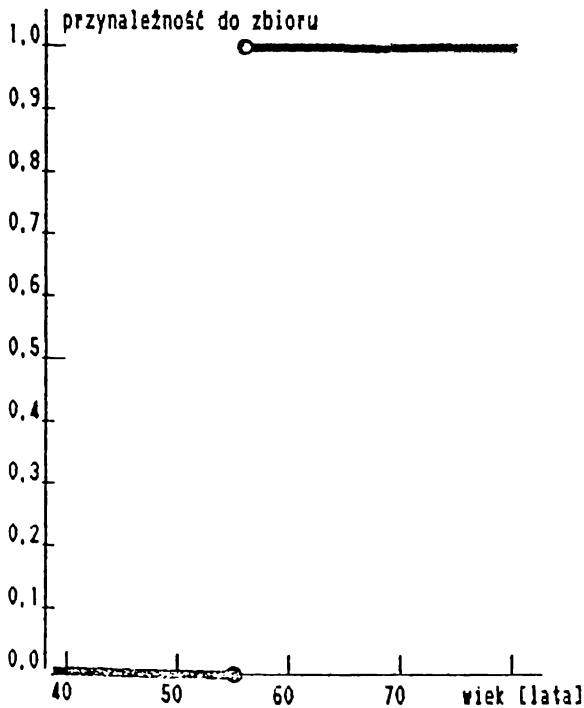
Większość nauk, obok posługiwania się dobrze zdefiniowanymi pojęciami, zmuszona jest również do używania określeń nieprecyzyjnych. Dotyczy to nawet tzw. nauk ścisłych, a w naukach medycznych jest wręcz regułą. Określenie "pacjent jest w podeszłym wieku" jest powszechnie uważane za słuszne w odniesieniu do osoby w wieku 90 lat, a w odniesieniu do 5 letniego dziecka każdy uzna je za absurdalne. Gdy jednak wiek pacjenta wynosi np. 55 lat, trudno jest jednoznacznie ocenić prawdziwość rozważanego stwierdzenia. Postępowanie medyczne oparte jest na prawidłowościach statystycznych, które sprawdzają się w większości przypadków, lecz nie są bezwzględnie słuszne. Mamy zatem do czynienia nie tylko z niepewnymi stwierdzeniami ale nawet reguły postępowania nie są niezawodne. Budowa systemów doradczych wymaga więc środków reprezentacji wiedzy i mechanizmów wnioskowania, które uwzględniałyby tego rodzaju niepewności. Wśród wielu prób formalnego ujęcia tych problemów szczególnie popularna jest teoria zbiorów rozmytych [47]. Poniżej przedstawione zostaną jedynie pewne intuicje związane z tą teorią; przystępne wprowadzenie w problematykę zbiorów rozmytych jest bo-



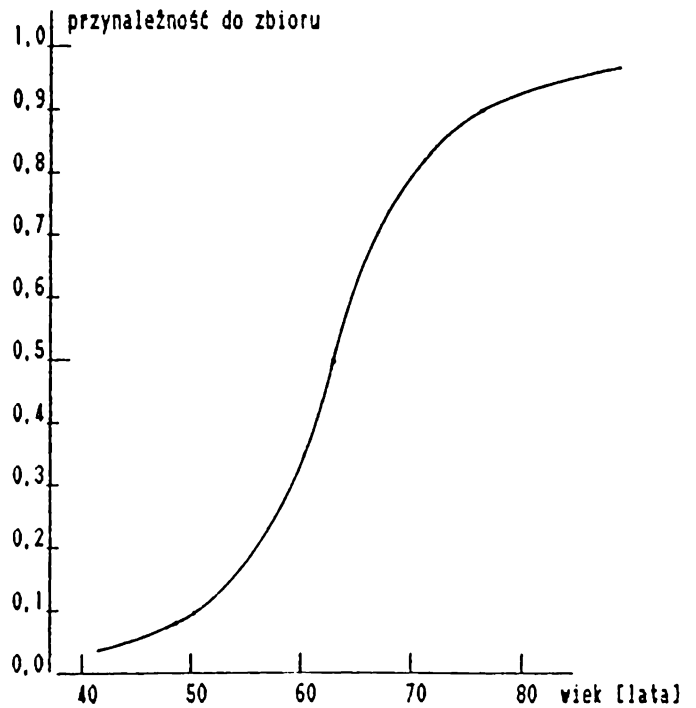
wiem łatwo dostępne [48].

### 2.6.1. Zbiory rozmyte.

W klasycznej teorii zbiorów podstawowym pojęciem jest przynależność elementu do zbioru. Np. REZERPINA należy do zbioru leków hipotensyjnych, a KOFEINA do tego zbioru nie należy. Zakłada się, że fakt przynależności elementu do zbioru jest jednoznacznie określony. Zbiór pacjentów w wieku powyżej 55 lat jest zbiorem w sensie klasycznym, natomiast zbiór pacjentów w podeszłym wieku - nie jest. Fakt przynależności elementów do zbioru można przedstawić przyporządkowując im liczbę 1 gdy należą do rozważanego zbioru i 0 gdy doń nie należą. Idea zbiorów rozmytych polega na przypisywaniu elementom dowolnych liczb z przedziału  $\langle 0,1 \rangle$ , określających tzw. stopień przynależności do zbioru. Przyporządkowanie elementom stopnia przynależności do zbioru można zilustrować graficznie;



Ryc. 2.2. Graficzna interpretacja klasycznego zbioru osób w wieku powyżej 55 lat.



Ryc. 2.3. Graficzna interpretacja rozmytego zbioru osób w podeszłym wieku.

ryc. 2.2 przedstawia klasyczny zbiór pacjentów w wieku powyżej 55 lat, natomiast ryc. 2.3 - możliwą propozycję rozmytego zbioru pacjentów w podeszłym wieku.

Z wykresu tego można odczytać, że przynależność osoby 50-letniej do zbioru osób w podeszłym wieku wynosi 10%, a osoby 75-letniej - 90% itp. Kształt tego wykresu został dobrany subiektywnie przez autora, można jednak przypuszczać, że propozycje innych osób nie odbiegałyby od niego w drastyczny sposób.

Analogicznie można rozważać rozmyte zbiory słusznych stwierdzeń lub reguł. W wielu systemach doradczych [34,35] zastosowano, wzorem propozycji Shortliffe'a i Buchanana zrealizowanej w systemie MYCIN [32,49], podobne podejście, w którym jednak dla każdego stwierdzenia rozważa się niezależnie dwie miary: stopień przekonania - MB (ang. *measure of belief*) i stopień wątpliwości - MD (ang. *measure of disbelief*). Ostateczną ocenę słuszności stwierdzenia, zwaną współczynnikiem pewności - CF (ang. *certainty factor*), stanowi różnica MB i MD:

$$CF = MB - MD \quad (2.1)$$

W odróżnieniu od funkcji przynależności zbioru rozmytego przybiera ona wartości z przedziału  $\langle -1, +1 \rangle$ , gdzie  $-1$  oznacza "na pewno nie", a  $+1$  - "na pewno tak".

### 2.6.2. Zastosowania w automatycznym wnioskowaniu.

Przyjęte przez autorów systemu MYCIN podejście jest wynikiem rozumowania heurystycznego, dla którego punktem wyjścia były rozważania probabilistyczne [49]. Uzyskane rezultaty można streścić podając własności miar MB i MD:

Rozważmy wnioskowanie: JEŻELI  $e$  TO  $h$ .

1.

Niech  $MB(h, e)$  i  $MD(h, e)$  oznaczają odpowiednio miarę przekonania i miarę wątpliwości wobec hipotezy  $h$ , gdy wiadomo na pewno, że s:

szne jest założenie e. Wtedy, dla dowolnych  $e_1, e_2$  zachodzi:

$$MB(h, e_1 \& e_2) = \begin{cases} 0, & \text{gdy } MD(h, e_1 \& e_2) = 1 \\ MB(h, e_1) + MB(h, e_2) - MB(h, e_1) \times MB(h, e_2), & \text{gdy } MD(h, e_1 \& e_2) < 1 \end{cases} \quad (2.2a)$$

$$MD(h, e_1 \& e_2) = \begin{cases} 0, & \text{gdy } MB(h, e_1 \& e_2) = 1 \\ MD(h, e_1) + MD(h, e_2) - MD(h, e_1) \times MD(h, e_2), & \text{gdy } MB(h, e_1 \& e_2) < 1 \end{cases} \quad (2.2b)$$

2°

Gdy  $e_1$  nie jest stwierdzeniem pewnym (tj.  $CF(e_1) < 1$ ), należy w powyższych wzorach  $MB(h, e_1)$  i  $MD(h, e_1)$  pomnożyć przez  $CF(e_1)$ .

3°

$$MB(e_1 \& e_2) = \min(MB(e_1), MB(e_2)) \quad (2.3)$$

$$MD(e_1 \& e_2) = \max(MD(e_1), MD(e_2))$$

Należy podkreślić, że konkretne wartości miar MB i MD są ustalane subiektywnie w czasie pozyskiwania wiedzy.

### 3. PROJEKT SYSTEMU

#### 3.1. Ogólne założenia projektu.

Projektując system informacji o leku przyjęto następujące, ogólne założenia:

- system przeznaczony jest przede wszystkim do wykorzystania przez lekarza lub farmaceutę klinicznego jako pomoc w jego codziennej pracy.
- informacje opracowywane są w niewielkim zespole i dotyczą wąskiej grupy leków.
- stworzone środki programowe powinny umożliwiać konstrukcję podobnych systemów dla innych grup leków.
- podstawowym zadaniem systemu jest dostarczenie wyczerpujących i aktualnych danych dotyczących wybranego leku oraz ułatwienie doboru właściwego leku w indywidualnym przypadku, zwłaszcza w złożonej sytuacji, gdy obok leczonego schorzenia pacjent posiada inne dolegliwości (stany fizjologiczne itp.) ograniczające stosowalność typowych leków.

#### 3.2. Wytyczne o charakterze informatycznym.

W wyniku analizy powyższych założeń sformułowano następujące postulaty, dotyczące informatycznych aspektów tworzonego systemu:

- konstrukcja systemu w postaci zintegrowanej, zespalającej baze danych o leku z odpowiednim systemem doradczym.
- relacyjny model bazy danych, złożona struktura danych, znaczna ilość informacji tekstowej.
- regułowa reprezentacja wiedzy w systemie doradczym; wnioskowanie niemonotoniczne w warunkach wiedzy niepewnej i niepełnej.
- konstrukcja środków programowych w sposób niezależny od informacji, które będą zawarte w systemie (ang. *shell*).

- możliwość modyfikacji bazy danych i bazy wiedzy przez administratora systemu.
- wygoda i łatwość opanowania przez użytkownika zasad posługiwania się programem.
- złożoność programu nie przekraczająca możliwości jednego programisty (tj. autora).

### 3.2.1. Uzasadnienie.

Zastosowanie "hybrydowej" konstrukcji, zespalającej bazę danych i system doradczy, przyjęto w wyniku rozważenia przydatności baz danych i stosowanych w systemach doradczych metod reprezentacji wiedzy dla przedstawienia informacji na temat leku. System informacji o leku można skonstruować na wzór systemów wyszukiwania dokumentów [50], stosowanych np. w informacji bibliograficznej, traktując jako dokument cały tekst opisujący właściwości leku. Dokument taki stanowi dla bazy danych systemu niepodzielną jednostkę; wyszukiwanie jest możliwe na podstawie nazwy leku lub słów kluczowych zawartych w odpowiednim tezaurusie. Posługiwanie się takim systemem jest raczej łatwe, łatwe jest też przygotowanie informacji w czasie tworzenia systemu. Natomiast precyzja wyszukiwania może być niska, niewielkie są również możliwości wykorzystania tak opracowanych danych przez inne programy użytkowe. W różnych ośrodkach w kraju (często nie związanych ze Służbą Zdrowia) podejmowano próby stworzenia systemów opartych o podobne zasady [14], najczęściej na bazie publikacji Podlewskich [10].

Innym rozwiązaniem może być przedstawienie informacji o leku w postaci bazy danych, w której różnym zagadnieniom związanym z lekiem odpowiadają różne atrybuty BD. Klasyczne bazy danych umożliwiają efektywne przechowywanie znacznych ilości informacji, jak również sprawny dostęp do tych informacji. Warunkiem wydajnego wykorzystania technologii baz danych jest istnienie wspólnego szablonu

nu dla opisu rozważanych obiektów. Znaczna część informacji o leku daje się ująć w taki sztywny schemat: dla każdego leku można podać nazwę międzynarodową, nazwy handlowe (zastrzeżone), grupę farmakologiczną, wskazania, przeciwwskazania itp. Korzystanie z takiego systemu jest trudniejsze, przygotowanie danych wymaga większej dyscypliny. Informacje są jednak lepiej usystematyzowane, a wyszukiwanie bardziej precyzyjne. Większe też są możliwości udostępnienia danych zawartych w bazie danych innym systemom. Ostatecznie oceniono ten wariant jako bardziej interesujący.

Niektóre komercyjne systemy informacji o leku funkcjonują jako taka baza danych. Jednakże w pewnych zagadnieniach więzy nałożone przez system bazy danych stają się za ciasne. Nie ma na przykład sensu wprowadzanie atrybutu GRUPA WILLIAMSA w sytuacji, gdy większość rozpatrywanych leków nie należy do żadnej z grup Williamsa. Problem tego typu ograniczeń płynących ze sztywnej struktury rekordów BD jest znany od dawna [51]. Interesującym sposobem złagodzenia rygorów charakterystycznego dla baz danych podejścia atrybut-wartość jest zaproponowane przez Chena [52] podejście jednostka-współzależność (ang. *entity-relationship*), które stało się punktem wyjścia dla tzw. *dedukcyjnych baz danych* [53, 54].

Bardziej elastycznymi środkami reprezentacji informacji dysponuje sztuczna inteligencja. Wygodnym sposobem przedstawienia obiektów o zróżnicowanej strukturze jest technika ram (zob. 2.3). Reprezentacja wiedzy w postaci ram wydaje się szczególnie atrakcyjna dla konstruktora systemu informacji w tak złożonej dziedzinie jak informacja o leku. Podejmowano już z powodzeniem tego rodzaju próby w zakresie interakcji leków i farmakologii [55]. Projektant dużego systemu informacji powinien poważnie przeanalizować tę reprezentację wiedzy. Jednakże, w przypadku systemu o mniejszym zakresie, ramy wydają się techniką zbyt kosztowną. Ich realizacja wymaga złożonego oprogramowania i (lub) znacznej pamięci operacyjnej.

Budując systemy doradcze rzadko zresztą korzysta się z samych ram - można np. regułową reprezentację wiedzy wesprzeć ramami, wykorzystując je nie tylko do opisu obiektów badanej dziedziny, lecz także do usystematyzowania zbioru reguł [45].

W związku z powyższymi uwagami, analizując swoje możliwości, zdecydowano wiedzę niezbędną do optymalnego doboru leku reprezentować w postaci reguł, natomiast pozostałe informacje powierzone relacyjnej bazie danych. W ten sposób można szybciej udostępnić użytkownikowi szeroki zakres informacji, oferując równocześnie automatyczne rozwiązywanie najczęściej spotykanego problemu (czyli doboru leku). Wyposażenie SZBD w pewne środki kontroli poprawności danych i zapewnienie komunikacji pomiędzy systemem doradczym a bazą danych może w przyszłości ułatwić zastosowanie bardziej zaawansowanych technik reprezentacji wiedzy.

Warto w tym miejscu zwrócić uwagę na głębokie związki pomiędzy logiką matematyczną, dającą podstawy teoretyczne wielu technikom reprezentacji wiedzy, a teorią relacyjnych baz danych. Znany algorytm Delobela-Casey'a [56] sprowadzenia zbioru relacji do 3PN jest oparty o teorię funkcji przełączających. Fagin [57] wskazuje na równoważność struktury zależności funkcjonalnych i rachunku zdań. Duże znaczenie teoretyczne ma propozycja języka zapytań do bazy danych opartego o rachunek predykatów I rzędu [1, 25, 29, 40]. Elementy jego praktycznej realizacji odnajdujemy w nowoczesnych językach zapytań, takich jak SQL, czy QUERY BY EXAMPLE. Pozwalają one na formułowanie pytań w formie czysto deklaratywnej. Interpretacja zapytań przez SZBD wymaga wyposażenia procesorów tych języków w odpowiednie mechanizmy wnioskowania, takie jak np. uzgadnianie (unifikacja).

Tak duża zbieżność problematyki dwu pozornie odległych dziedzin informatycznych, jakimi są bazy danych i sztuczna inteligencja, stanowi dodatkową zachętę dla zbadania możliwości wspólnego ich u-

względnienia w omawianym projekcie.

Postulat relacyjnego modelu danych wynika przede wszystkim z prostoty tego modelu. Dokładniejsza analiza przeprowadzona jest w rozdziale omawiającym projekt SZBD (punkt 3.3).

Wiedza z zakresu nauk medycznych (a więc także farmakologiczna) ma zdecydowanie niepewny charakter. Ten sam lek podany na pozór podobnym pacjentom może u każdego z nich odnieść inny skutek, np. przynosząc w jednym przypadku znaczną poprawę, w drugim nie dając wyraźnych efektów, a w jeszcze innym powodując niebezpieczne komplikacje. Lekarz niejednokrotnie musi podjąć decyzję, pomimo że nie posiada wszystkich, jego zdaniem istotnych informacji. Ponadto, lek przyjęty w wyniku wstępnych rozważań, może być po głębszej analizie odrzucony. Naśladowanie przez program komputerowy rozumowania doświadczonego lekarza wymaga więc zdolności tzw. wnioskowania niemonotonicznego [58], czyli takiego, w którym możliwe jest odrzucenie, w świetle nowych informacji, uprzednio uzyskanych wniosków. Wnioskowanie to powinno być prowadzone zgodnie z zasadami rozumowania przybliżonego, omówionymi w punkcie 2.6.2, gdzie podano również standardowy algorytm [49]. Lekarz-specjalista dobiera lek na drodze specyficznego, w znacznej mierze intuicyjnego rozumowania. W rozumowaniu tym, z wyczuciem płynącym z wiedzy i doświadczenia, bilansuje on wskazania i przeciwwskazania rozważanych leków. Powstaje pytanie, czy zastosowanie standardowego algorytmu rozumowania przybliżonego pozwoli na zadowalającą symulację tego procesu decyzyjnego oraz, czy potrzebne intuicje da się wyrazić w formalizmie reguł i ich stopni pewności?

Udzielenie, na drodze empirycznej, odpowiedzi na to pytanie jest jednym z głównych zadań, jakie stawia przed sobą niniejsza praca.

Konstrukcja programu w sposób uzależniony od rodzaju informacji, które mają być w nim zawarte (np. od wybranej grupy leków), byłaby rozwiązaniem bardzo nieopłacalnym. Taka zależność pojawia się wte-



dy, gdy programista wyraża w języku programowania informacje specyficzne dla aktualnego zastosowania. Oczywiście wykorzystanie takiego systemu do innego celu (np. do przedstawienia innej grupy leków), a nawet tylko modyfikacja pewnych założeń wyjściowego systemu wymaga zmian w programie. W praktyce oznacza to, że (w najlepszym przypadku) tylko autor programu może go użyć do celu innego niż pierwotny i to zwykle przy niemałym nakładzie pracy. Dlatego znacznie lepiej jest cechy specyficzne dla konkretnego zastosowania umieszczać poza tekstem programu, umożliwiając ich redakcję i modyfikację bez potrzeby jakichkolwiek zmian w samym programie. W ten sposób powstaje "pusty system", wyposażony w narzędzia umożliwiające "napelnienie" go treścią, zależną od konkretnego zastosowania. Biorąc pod uwagę nieuchronność modyfikacji założeń tworzonego systemu oraz chęć wykorzystania go do opracowania innych grup leków, niezależność programu od danych staje się oczywistym postulatem. Takie rozwiązanie ułatwia pozostawienie możliwości dokonywania modyfikacji także użytkownikom (administratorowi) systemu, pozwalając im na dopasowanie go do swych potrzeb.

Ostatnie dwa postulaty nie wymagają komentarza.

### 3.3. Założenia projektu Systemu Zarządzania Bazą Danych.

Projektując SZBD przyjęto następujące postulaty:

- relacyjny model danych.
- niezależność logicznej struktury bazy danych od realizacji fizycznej.
- możliwość przedstawienia BD w postaci zbioru relacji (dla umożliwienia normalizacji do co najmniej 3PN).
- szeroki wachlarz dostępnych typów danych, sposobów ich kodowania i organizacji fizycznych plików z danymi.
- wygodne środki projektowania struktury bazy danych i wprowadzania danych.
- wygodny i łatwy do opanowania sposób formułowania zapytań i poleceń dla systemu, przejrzysta prezentacja odpowiedzi.

Wybór relacyjnego modelu danych podyktowany był następującymi zaletami tego modelu:

- prostota logicznego obrazu danych.
- łatwość modyfikacji schematu.
- możliwość przedstawienia odrębnych zagadnień w postaci osobnych relacji.

Wady modelu relacyjnego (trudność osiągnięcia wysokiej efektywności i niskiej nadmiarowości) w przypadku stosunkowo niewielkiego rozmiaru bazy danych mają znaczenie drugorzędne; poza tym dążono do zmniejszenia ich wpływu.

Postulat niezależności fizycznej i logicznej wynika z nieuniknionych zmian w organizacji, zarówno logicznej jak i fizycznej, zachodzących w czasie rozwoju projektu. Byłoby niekorzystnie, gdyby każda zmiana w jednej z tych organizacji musiała pociągać zmiany w innej, lub co gorsza wymuszała zmiany w programach użytkowych.

Przedstawienie BD w postaci zbioru relacji jest potrzebne nie tylko ze względu na zmniejszanie redundancji na drodze normalizacji. Po-

dział na relacje pozwala także wyodrębnić powiązane tematycznie grupy atrybutów i w ten sposób ułatwić użytkownikowi orientację w schemacie logicznym BD.

Charakter przewidywanego zastosowania wskazywał na potrzebę korzystania co najmniej z takich typów danych, jak teksty dowolnej długości, liczby całkowite i rzeczywiste. W projekcie przyjęto jednak bogatszy zestaw typów. Pełna lista przewidzianych typów danych jest następująca:

- liczby całkowite
- liczby rzeczywiste
- łańcuchy znaków określonej długości
- łańcuchy znaków dowolnej długości
- typy wyliczeniowe
- zbiory (w sensie teoriomnogościowym)

*Liczby całkowite i rzeczywiste oraz łańcuchy znakowe określonej długości* należą do podstawowych typów danych stosowanych w BD, zostały zatem przewidziane i w tym projekcie.

*Teksty (łańcuchy znaków dowolnej długości)* musiały wejść do projektu z uwagi na znaczną ilość informacji opisowych, np. takich jak opis działań ubocznych czy przeciwwskazań dla leku.

Potrzeba wprowadzenia *typów wyliczeniowych* wynika z istnienia atrybutów o niewielkiej dziedzinie (tj. mogących przyjmować jedynie kilka wartości); np. zbiór wartości atrybutu UKŁAD (używanego w relacji DZIAŁANIE UBOCZNE) składa się z następujących określeń:

( POKARMOWY, NERWOWY, ODDECHOWY, KRĄŻENIA, DOKREWNY, WYDALNICZY, ROZRODCZY, KREW, SKÓRA ).

Atrybut UKŁAD w konkretnym opisie działania leku na dany układ przyjmuje zawsze jedną z tych wartości, a zadaniem SZBD jest czuwanie nad prawidłowością wartości przybieranych przez ten atrybut (tj. sprawdzanie, czy pochodzą one z przedstawionego zbioru). Po-

traktowanie tego atrybutu jako danej znakowej uniemożliwiłoby taką kontrolę. Istnieją jednakże sytuacje jeszcze bardziej złożone. W wykorzystywanej przez system doradczy relacji zawierającej wywiad lekarski występuje atrybut SCHORZENIA TOWARZYSZĄCE. Typ wartości tego atrybutu należy określić jako *zbiór* elementów wyliczeniowego typu CHOROBY, ponieważ jeden pacjent może jednocześnie cierpieć na kilka schorzeń.

Uznano również za pożądane istnienie typu "*ciąg elementów*", a zwłaszcza takiego jego wariantu, który pozwalałby na efektywne przechowywanie odnośników do literatury; niektóre pozycje literaturowe dotyczą wielu leków i każdorazowe wpisywanie takiego odnośnika jest marnotrawstwem pamięci.

Dostępność różnych organizacji fizycznych jest potrzebna dla efektywnego funkcjonowania systemu. Dziedziny atrybutów tworzących klucze relacji są różne i dla wydajnego wyszukiwania i przechowywania wartości atrybutów identyfikowanych przez te klucze należy używać różnych organizacji plików bazy danych. Uznano za konieczne uwzględnienie organizacji rozproszonej, bezwzględnej (klucz równy numerowi krotki) i sekwencyjnej [2, 59, 60, 61]. W dalszej perspektywie pożądane jest włączenie organizacji indeksowo-sekwencyjnej, rozszerzalnej rozproszonej (ang. *extendible hashing*) [62] i innych. Wygoda środków projektowania jest potrzebna dla wydajnej ewolucji projektu, natomiast postulat łatwego formułowania zapytań wynika z zamiaru udostępnienia systemu użytkownikom nieprofesjonalnym.

### 3.3.1. Projekt języka zapytań.

Głównym postulatem przyjętym w czasie projektowania języka była prostota formułowania zapytań i minimalizacja możliwości popełnienia błędu. W projekcie wykorzystano pewne idee języka QUERY BY EXAMPLE. Założono m.in. całkowitą nieproceduralność języka zapytań, możliwość realizacji zapytań odpowiadających projekcji, selek-

konania operacji połączenia na kilku relacjach. Projektowany język zapewnia więc realizację podstawowych operacji algebry relacyjnej.

### 3.4. Założenia projektu programu pozwalającego na tworzenie systemów doradczych.

Projektując program przyjęto następujące postulaty:

- wiedza wyrażana w postaci reguł wnioskowania.
- reprezentacja stwierdzeń w postaci obiekt-atrybut [7].
- możliwość rozumowania w warunkach niepełnej i niepewnej wiedzy, zdolność operowania pojęciami rozmytymi.
- wnioskowanie wstecz.
- istnienie mechanizmów objaśniania.
- możliwość komunikacji z bazą danych, wykonywania obliczeń i wyświetlania komunikatów (ostrzeżeń itp.) w czasie wnioskowania.
- sprawne mechanizmy pozyskiwania wiedzy, w szczególności wygodny edytor reguł wnioskowania.
- wygodny i łatwy do opanowania sposób formułowania zapytań i poleceń dla systemu.

#### 3.4.1. Uzasadnienie.

Wybór reprezentacji wiedzy w postaci reguł wnioskowania został już uzasadniony w punkcie 3.2.1. Szczegółowy projekt tej reprezentacji zostanie omówiony w punkcie 4.2.5. Reprezentacja stwierdzeń w postaci obiekt-atrybut wydaje się wystarczająca w tworzonym systemie (por. 3.4.2.).

Biorąc pod uwagę docelowe zastosowanie systemu, konieczne staje się założenie o możliwości rozumowania w warunkach niepełnej i niepewnej wiedzy, taką bowiem jest wiedza medyczna czy farmakologiczna. Również ocena pewnych faktów, np. tego, czy pacjent mający 60 lat może być uznany za osobę w podeszłym wieku nie jest jednoznaczna i do opisu takich faktów dobrze jest posłużyć się metodami teorii zbiorów rozmytych.

Tworzony system ma służyć przede wszystkim do oceny przydatności rozważanych leków, wykorzystując tylko te dane na temat pacjenta,

które są potrzebne. Skłania to do wyboru wnioskowania wstecz jako stosowanego przez system schematu rozumowania, gdyż znane są hipotezy (np. "Propranolol jest wskazanym lekiem") i należy orzec o ich prawdziwości, w razie potrzeby uzupełniając dane. Zastosowanie wnioskowania w przód wymagałoby od lekarza każdorazowo podania wszystkich znanych faktów (w tym z pewnością wielu nieistotnych), a przeoczenie posiadanej, istotnej informacji, mogłoby w końcowym efekcie dawać fałszywe wnioski.

Istnienie mechanizmów objaśniania jest bardzo pożądane, gdyż użytkownik może być zainteresowany, jakie przyczyny spowodowały wybór danego leku lub odrzucenie innego. Sprawny mechanizm objaśniania otwiera przed systemem niezwykle atrakcyjną szansę wykorzystania go do celów dydaktycznych.

Uzyskanie pewnych, potrzebnych w czasie wnioskowania informacji może wymagać wykonania obliczeń lub odczytania wymaganych danych z bazy danych (np. odczyt odpowiednich danych z wywiadu lekarskiego).

W czasie rozumowania SD może osiągnąć pewne uboczne, ale ważne wnioski. W związku z tym pojawia się potrzeba wyświetlania odpowiednich komunikatów. Takim komunikatem może być np. ostrzeżenie o podejrzanym zatruciu glikozydami naparstnicy.

Postulat posiadania wygodnego edytora reguł wynika oczywiście stąd, że tworzenie SD wymaga częstych poprawek i rozszerzeń, których dokonywanie nie powinno być uciążliwe.

Podobnie, wymagania odnośnie łatwości posługiwania się systemem, są nieodzownym warunkiem aby mógł on w ogóle znaleźć użytkowników.

### 3.4.2. Projekt reprezentacji stwierdzeń.

Podstawowe znaczenie dla procesu wnioskowania mają stwierdzenia (fakty). Ich budowę opisuje prosta gramatyka:

<stwierdzenie> ::= <część podmiotu> <orzeczenie> <określenie>

Z każdym określeniem skojarzone są dwie możliwe postaci orzeczenia - twierdząca i przecząca. Przy pewnej dyscyplinie tak tworzone stwierdzenia mogą być na ogół poprawnymi polskimi zdaniami.

**Przykład:**

<część podmiotu> ::= propranolol  
<orzeczenie> ::= jest  
<określenie> ::= wskazanym lekiem

Zaproponowana reprezentacja stwierdzeń posiada możliwości zbliżone do często stosowanego rozwiązania <obiekt, atrybut, stopień pewności> [7], przy czym obiektowi odpowiada część podmiotu, a atrybutowi - określenie. Informacja o tym, czy obiektowi należy przypisać dany atrybut zawarta jest w fakcie wyboru twierdzącej lub przeczącej wersji orzeczenia. Składniki stwierdzenia (<część podmiotu>, <orzeczenie>, <określenie>) stanowią informację symboliczną, dla której należało znaleźć odpowiednią reprezentację.

**3.4.3. Wykorzystanie środków SZBD dla organizacji bazy wiedzy.**

Biorąc pod uwagę, że omawiany system tworzony jest w środowisku istniejącego już SZBD, zdecydowano (eksperymentalnie) całą informację symboliczną (reguły, stwierdzenia itp.) zorganizować w formie klasycznej bazy danych. Rozwiązanie to przyjęto, kierując się następującymi przesłankami:

1\*. Znika potrzeba tworzenia nowych środków do reprezentacji informacji symbolicznej. Informacja ta i tak musi być przechowywana w pamięci zewnętrznej (dyskowej), nawet jeśli podczas funkcjonowania systemu wprowadza się ją do pamięci operacyjnej. Skoro już istnieją środki dla organizacji takiej informacji na nośniku zewnętrznym, to naturalne wydaje się ich wykorzystanie.

2\*. W przypadku eksploatacji bazy wiedzy w pamięci operacyjnej rozmiary tej pamięci narzucają, niekiedy dość drastyczne, ograniczenia na ilość dostępnej wiedzy (ilość reguł itp). Przeniesienie inform-



macji symbolicznej do pamięci zewnętrznej praktycznie usuwa te ograniczenia.

3\*. Realizacja systemu w środowisku SZBD ułatwia systemowi korzystanie z istniejących w tym środowisku baz danych, co stwarza dodatkowe możliwości, np. w automatyzacji procesu pozyskiwania wiedzy.

4\*. Tworzenie bazy wiedzy jest procesem iteracyjnym. Dostępność operacji wstawiania i kasowania (INSERT/DELETE) ułatwia konstrukcję edytora umożliwiającego wprowadzanie zmian w bazie wiedzy (np. dodawanie lub usuwanie reguł).

5\*. Nie ma żadnych ograniczeń dotyczących znaków, które mogą być użyte w symbolach. W szczególności można używać odstępów, których nie trzeba (jak w wielu znanych systemach) zastępować znakiem podkreślenia - piszemy więc UPOSLDZENIE FUNKCJI WATROBY zamiast UPOSLDZENIE\_FUNKCJI\_WATROBY.

Oczywiście należy się liczyć z wadami przyjętego rozwiązania, w szczególności może ono być nieefektywne czasowo. Ustalenie, czy przeważą wady czy zalety może być jednak uzyskane jedynie na drodze eksperymentalnej.

Techniczne szczegóły dotyczące wykorzystania technologii relacyjnych BD do realizacji regułowej bazy wiedzy systemu doradczego omówione zostaną w punktach 4.3.3 i 4.3.4.

### 3.5. Projekt bazy danych o lekach krążeniowych.

#### 3.5.1. Schemat logiczny.

Schemat logiczny opracowano zgodnie z wytycznymi zawartymi w pracy Wójcik-Jawień [23], równocześnie starając się zapewnić normalizację relacji bazy danych do co najmniej 3PN. Projektowana baza danych składa się z następujących relacji:

- PODSTAWOWE WŁASNOŚCI
- DZIAŁANIE UBOCZNE
- NAZWY
- FARMAKOKINETYKA
- DAWKOWANIE
- INTERAKCJE
- PIŚMIENNICTWO
- UWAGI

Podział bazy danych na relacje: DZIAŁANIE UBOCZNE, NAZWY, FARMAKOKINETYKA i INTERAKCJE jest wynikiem normalizacji; wyróżnienie relacji: PODSTAWOWE WŁASNOŚCI, DAWKOWANIE, PIŚMIENNICTWO i UWAGI nastąpiło tylko ze względu na logiczną niezależność tych zagadnień.

Rycina 3.1 przedstawia strukturę każdej relacji. Wyszczególniono wszystkie atrybuty. Składniki klucza przedstawiono w podwójnych ramkach.

#### PODSTAWOWE WŁASNOŚCI:

Nazwa międzynarodowa	Nazwa międzynarodowa w brzmieniu polskim	Nazwa chemiczna	Wzór sumaryczny	pH	pK <sub>a</sub>	Temperatura topnienia	Warunki przechowywania
----------------------	--	-----------------	-----------------	----	-----------------	-----------------------	------------------------

Grupa farmakologiczna	Mechanizm działania	Działanie na inne układy	Wskazania	Przeciwwskazania względne	Przeciwwskazania bezwzględne
-----------------------	---------------------	--------------------------	-----------	---------------------------	------------------------------

Ryc. 3.1. Schemat logiczny bazy danych o leku

*cd. na stronie następnej*

DZIAŁANIE UBOCZNE:

Nazwa międzynarodowa	Układ	Opis objawów
----------------------	-------	--------------

NAZWY:

Nazwa międzynarodowa	Nazwa zastrzeżona	Firma farmaceutyczna	Kraj
----------------------	-------------------	----------------------	------

FARMAKOKINETYKA:

Nazwa międzynarodowa	Parametr farmakokinetyczny	Opis
----------------------	----------------------------	------

INTERAKCJE:

Nazwa międzynarodowa LEK 1	Nazwa międzynarodowa LEK 2	Rodzaj interakcji	Opis interakcji
-------------------------------	-------------------------------	-------------------	-----------------

DAWKOWANIE:

Nazwa międzynarodowa	Opis dawkowania	Dostępność w kraju
----------------------	-----------------	--------------------

PIŚMIENICTWO:

Nazwa międzynarodowa	Wykaz piśmiennictwa
----------------------	---------------------

UWAGI:

Nazwa międzynarodowa	Uwagi
----------------------	-------

We wszystkich relacjach składnikiem klucza jest atrybut 'Nazwa międzynarodowa'. Jest on jedynym składnikiem klucza (czyli kluczem prostym) relacji PODSTAWOWE WŁASNOŚCI, DAWKOWANIE, PIŚMIENICTWO i UWAGI. Relacje: DZIAŁANIE UBOCZNE, NAZWY, FARMAKOKINETYKA i INTERAKCJE posiadają klucze złożone. Działania uboczne są posegregowane według układów, ze strony których widoczne są ich objawy. Taką segregację osiągnięto przez wprowadzenie drugiego składnika klucza, tj. atrybutu 'Układ'. Podobnie w opisie farmakokinetyki drugim składnikiem klucza jest atrybut 'Parametr farmakokinetyczny'. W relacji NAZWY wszystkie atrybuty wchodzi w skład klucza, gdyż dopiero podanie ich wszystkich ('Nazwa międzynarodowa', 'Nazwa zastrzeżona', 'Firma', 'Kraj') jednoznacznie identyfikuje preparat (zdarza się, że tej samej nazwy zastrzeżonej używają różne firmy; również nie można z całkowitą pewnością wykluczyć użycia jednej nazwy zastrzeżonej dla leków różniących się nazwą międzynarodową<sup>2</sup>; wreszcie jedna firma może produkować dany preparat w kilku krajach pod tą samą nazwą<sup>3</sup>).

W relacji INTERAKCJE klucz jest trójskładnikowy: dwa pierwsze jego składniki (nazwane tu 'LEK 1' i 'LEK 2'; w istocie pod hasłem 'LEK 2' może kryć się grupa farmakologiczna lub pokarm) określają pomiędzy jakimi środkami zachodzi interakcja (np. DIGOKSYNA - jony  $Ca^{++}$ ), trzeci składnik określa rodzaj interakcji (tj. informuje, czy jest to interakcja typu lek-lek, lek-grupa farmakologiczna, czy lek-pokarm). W przypadku interakcji typu lek-lek składniki klucza 'LEK 1' i 'LEK 2' są symetryczne (zob. 1.3.4.), gdyż w ogólnym przypadku interakcja opisuje nie tyle wpływ jednego leku na działanie drugiego, lecz raczej wypadkowy efekt ich równoczesnego podania. W takim razie interakcje DIGOKSYNA - CHINIDYNA i CHINIDYNA - DIGOKSYNA są identyczne i w związku z tym odpowiada im tylko jeden zapis w bazie danych. Zależność symetrii składników klucza od wartości innego składnika jest cechą niekorzystną. Można by jej

uniknąć, rozbijając relację INTERAKCJE na dwie - jedną z interakcjami typu LEK-LEK i symetrycznym kluczem oraz drugą, zawierającą pozostałe interakcje, ze zwykłym kluczem. Rozwiązania tego jednak nie zastosowano, ze względu na dążenie do zachowania prostoty schematu (a co za tym idzie łatwości użytkowania) tworzonej bazy danych, mimo iż prowadzi to do pewnych utrudnień w projekcie i konstrukcji programu.

### 3.5.2. Organizacja fizyczna.

Dla wszystkich relacji z kluczem prostym przewidziano mieszającą organizację pliku. Gwarantuje to krótki czas wyszukiwania, niezależnie od rozmiaru bazy danych i (w przeciwieństwie do organizacji indeksowych) nie zajmuje pamięci operacyjnej (co miało zasadnicze znaczenie w początkowej fazie realizacji projektu). Wady tej organizacji to ustalony rozmiar pliku i przypadkowa kolejność rekordów w pliku. Organizacja mieszająca nie nadaje się do zastosowań, w których ilość rekordów w BD gwałtownie się zmienia; jednakże w ciągu roku przybywa niewiele nowych leków krążeniowych, a niektóre istniejące wychodzą z użycia - zatem wzrost bazy danych jest raczej powolny. Ponadto istnieje modyfikacja tej metody dostępu odporna na zmiany rozmiaru bazy danych [62], choć wymagająca pośredniego indeksu. Większym mankamentem jest przypadkowa (niealfabetyczna) kolejność rozmieszczenia kluczy, gdyż wyniki przeszukiwania będą prezentowane niealfabetycznie, co niekiedy może być niewygodne. W przyszłości przewiduje się rozważenie innych organizacji.

W relacjach o kluczu złożonym pierwszy etap dostępu jest również realizowany zgodnie z organizacją mieszającą (wg wartości pierwszego składnika klucza). Lokalizacja wg pozostałych składników klucza przebiega zgodnie z organizacją sekwencyjną lub absolutną.

Organizację absolutną zastosowano w relacjach DZIAŁANIE UBOCZNE (atrybut 'Układ'), FARMAKOKINETYKA (atrybut 'Parametr farmakokine-



tyczny') i INTERAKCJE (atribut 'Rodzaj interakcji'). Użycie dla tych atrybutów organizacji absolutnej wynika stąd, że wszystkie one mają kilkuelementową dziedzinę (typy tych atrybutów są typami wyliczeniowymi).

Organizację sekwencyjną zastosowano dla następujących atrybutów: 'Nazwa zastrzeżona', 'Firma', 'Kraj' (relacja NAZWY) oraz 'LEK 2' (relacja INTERAKCJE), gdyż dziedziny tych atrybutów są duże.

### 3.6. Projekt systemu doradczego dla doboru leku krążeniowego.

Przy projektowaniu systemu doradczego kierowano się wytycznymi zawartymi w rozdziale 4 pracy Wójcik-Jawień [23], biorąc równocześnie pod uwagę możliwości programu komputerowego. Projekt obejmuje logiczny schemat bazy danych (ściślej: pojedynczej relacji) z ogólnym wywiadem lekarskim, sformułowanie celów, które mają być osiągnięte i wyszczególnienie celów pośrednich.

#### 3.6.1. Schemat logiczny relacji WYWIAD.

Schemat logiczny relacji przeznaczonej do przechowywania wywiadu lekarskiego przedstawia ryc. 3.2.

WYWIAD:

Symbol identyfikacyjny	Nazwisko	Imię	Płeć	Wiek	Wzrost	Waga	Tętno	Ciśnienie skurczowe	Ciśnienie rozkurczowe	Schorzenie leczone
------------------------	----------	------	------	------	--------	------	-------	---------------------	-----------------------	--------------------

Ryc. 3,2, a.

Schorzenia towarzyszące	Schorzenia przebyte	Glukoza	Cholesterol	Kreatynina	Potas	Kwas moczowy
-------------------------	---------------------	---------	-------------	------------	-------	--------------

Ciśn. skurcz. pomiar I	Ciśn. skurcz. pomiar II	Ciśn. skurcz. pomiar III	Ciśn. rozkurcz. pomiar I	Ciśn. rozkurcz. pomiar II	Ciśn. rozkurcz. pomiar III
------------------------	-------------------------	--------------------------	--------------------------	---------------------------	----------------------------

Ryc. 3,2, b.

Klirens kreatyniny	Waga należna	Stadium nadciśnienia	Postać nadciśnienia	Charakter zaburz. psych.	Wydolność ukł. krążenia
--------------------	--------------	----------------------	---------------------	--------------------------	-------------------------

Okres nadciśnienia	Postać dusznicy	Stopień bloku AV	Rodzaj arytmii	Trymestr ciąży	Podawane leki	Aktywność reniny
--------------------	-----------------	------------------	----------------	----------------	---------------	------------------

Ryc. 3,2. Schemat logiczny relacji z wywiadem lekarskim,

- a) Dane podstawowe,
- b) Dane uzupełniające.

Atrybuty przedstawione w części a ryc. 3.2 wchodzi w skład podstawowego wywiadu i oczekuje się, że dla dowolnego pacjenta są one

określone i mają sensowne wartości. Atrybuty zajmujące część bryliny stanowią dane uzupełniające, tj. są istotne tylko w pewnych sytuacjach, np. wartości trzykrotnego pomiaru ciśnienia są potrzebne tylko w przypadku, gdy pacjent cierpi na nadciśnienie tętnicze, natomiast rubryka trymestr ciąży ma sens tylko u kobiet ciężarnych. Typy atrybutów są następujące:

- łańcuchy znaków - Symbol identyfikacyjny, Nazwisko, Imię;
- liczby całkowite - Wiek, Wzrost, Tętno, Ciśnienie skurczowe oraz Ciśnienie rozkurczowe i ich kolejne pomiary;
- liczby rzeczywiste - Waga, Glukoza, Cholesterol, Kreatynina, Potas, Kwas moczowy, Klirens kreatyniny, Waga należąca oraz średnie wartości ciśnienia skurczowego i rozkurczowego
- typy wyliczeniowe - Płeć, Stadium nadciśnienia, Postać nadciśnienia, Wydolność układu krążenia, Okres nadciśnienia, Stopień bloku AV, Trymestr ciąży, Aktywność reniny.
- zbiory elementów - Schorzenia leczone, Schorzenia towarzyszące, Schorzenia przebyte, Charakter zaburzeń psychicznych, Postać dusznicy, Rodzaj arytmii, Podawane leki.

Wykaz składników typów wyliczeniowych przedstawia tabela 3.1.



Tabela 3.1. Wykaz atrybutów o typach wyliczeniowych.

Atrybuty:	Nazwa typu:	Składniki:
Płeć	_Płeć	MEŹCZYŻNA KOBIEȚA
Schorzenie leczone	_Choroby_lecz	ZABURZENIA RYTMU SERCA NADCIŚNIENIE TĘTNICZE SAMOISTNE DUSZNICIA BOLESNA
Schorzenia towarzyszące Schorzenia przebyte	_Choroby	PRZEWLEKŁA NIEWYDOLNOŚĆ KRAŻENIA ZAWAŁ SERCA UDAR MÓZGU UPOŚLEDZENIE KRAŻENIA OBWODOWEGO STANY BRONCHOSPASTYCZNE CUKRZYCA ŻÓLTACZKA NIEWYDOLNOŚĆ WATROBY CHOROBA WRZODOWA NIEWYDOLNOŚĆ NEREK NADCZYNNOŚĆ TARCZYCY ZABURZENIA PSYCHICZNE MYASTHENIA GRAVIS
Stadium nadciśnienia	_Stadia	STADIUM I STADIUM II STADIUM III
Postać nadciśnienia	_Postaci	ŁAGODNE UMIARKOWANE CIĘŻKIE
Charakter zaburzeń psychicznych	_Zab_psych	CIĘŻKA DEPRESJA UMIARKOWANA DEPRESJA INNÉ ZABURZENIA
Wydolność układu krążenia	_NYHA	WYDOLNE NYHA I ST. NYHA II ST. NYHA III ST. NYHA IV ST.
Okres nadciśnienia	_Okresy_nadc	OKRES I OKRES II OKRES III
Postać dusznicy	_Dusznice	DUSZNICA WYSIŁKOWA DUSZNICA PRINZMETALA DUSZNICA NIESTABILNA
Stopień bloku AV	_Stopnie_bloku	BLOK I STOPNIA BLOK II STOPNIA BLOK III STOPNIA

*ciąg dalszy na stronie następanej*

Tabela 3.1. (ciąg dalszy).

Rodzaj arytmii	_Arytmie	PRZYSPIESZENIE ZATOKOWE EKSTRASYSTOLIA PRZEDSIONKOWA NAPADOWY CZĘSTOSKURCZ NADKOMOROWY CZĘSTOSKURCZ PRZEDSIONKOWY Z BLOKIEM NIEWYDOLNOŚĆ WĘZŁA SA TRZEPOTANIE PRZEDSIONKOW NAPADOWE MIGOTANIE PRZEDSIONKÓW UTRWALONE MIGOTANIE PRZEDSIONKÓW WIELOKSZT. CZĘSTOSKURCZ PRZEDSIONKOWY EKSTRASYSTOLIA WĘZŁOWA NIENAPADOWY CZĘSTOSKURCZ WĘZŁOWY EKSTRASYSTOLIA KOMOROWA NAPADOWY CZĘSTOSKURCZ KOMOROWY ZESPÓŁ WPW
Trymestr ciąży	_Trymestry	BRAK CIĄŻY TRYMESTR I TRYMESTR II TRYMESTR III
Podawane leki	_Leki	AMINOFILINA CHINIDYNA DIZOPIRAMID KLOFIBRAT PROKAINAMID GLIKOZYDY NAPARSTNICY TRÓJCYKLICZNE LEKI PRZECIWDOPRESYJNE
Aktywność reniny	_Akt_reniny	W NORMIE OBNIŻONA PODWYŻSZONA NIEZNANA

### 3.6.2. Organizacja fizyczna.

Dla relacji WYWIAD przewidziano organizację o dostępie przypadkowym, realizowanym wg klucza 'Symbol identyfikacyjny'. Nie ma potrzeby stosowania specjalnego kodowania dla żadnego z atrybutów.

### 3.6.3. Cele (etapy) wnioskowania.

Głównym celem wnioskowania systemu jest wybór, w zależności od podstawowego schorzenia ('Schorzenie leczone'), optymalnego leku hipotensyjnego, przeciwaritmicznego lub przeciwdusznicy. Wiele spośród rozważanych leków ma zastosowanie w więcej niż jednym schorzeniu, np.  $\beta$ -blokery mogą być stosowane zarówno w nadciśnieniu, jak i w zaburzeniach rytmu serca, a także w dusznicy bolesnej. Z kolei nierzadkie są przypadki pacjentów, u których występują

wszystkie trzy rozważane choroby serca. Mając na uwadze maksymę głoszącą, że *leczyć należy pacjenta a nie chorobę*, chciałoby się główny cel sformułować w postaci pytania: "Co jest wskazanym lekiem?". Rozwiązanie takie jest jednak niekorzystne z następujących powodów:

1'. System, w najprostszycch nawet przypadkach, analizowałby wszystkie leki, także takie, które w konkretnym przypadku nie mają zastosowania. Np. u pacjenta z dusznicą bolesną ale nie cierpiącego na nadciśnienie przeprowadzana byłaby analiza przeciwwskazań dla diuretyków - leków, stosowanych wyłącznie w nadciśnieniu. Oczywiście, ze względu na brak wskazań, zostałyby one ostatecznie wyeliminowane, ale zajmowanie się nimi byłoby ewidentną stratą czasu i prowadziłyby do niepotrzebnych, a niekiedy i bezsensownych pytań zadawanych użytkownikowi przez system.

2'. W złożonych przypadkach trzeba stosować kilka leków. Lista leków uzyskana w wyniku odpowiedzi na pytanie "Co jest wskazanym lekiem?" zawierałaby leki uszeregowane wg stopnia przydatności u pacjenta. Znajdowałyby się na niej obok siebie zarówno leki podobne, które można stosować zamiennie, jak i leki z różnych grup, które trzeba podawać łącznie z innymi. Z takiej listy lekarz musiałby dopiero wyłuskać właściwy zestaw leków.

W świetle powyższych uwag zdecydowano wyróżnić trzy główne cele, które można wyrazić pytaniami:

- Co jest wskazanym lekiem przeciwarrytmicznym?
- Co jest wskazanym lekiem hipotensyjnym?
- Co jest wskazanym lekiem przeciwdusznicyowym?

W przypadku pacjenta o kilku schorzeniach w rubryce 'Schorzenie leczone', wnioskowanie wymaga osiągnięcia tylu celów, ile jest schorzeń, choć może się okazać, że jeden lek będzie realizacją więcej niż jednego celu.

Ogólne sformułowanie w rodzaju "Co jest wskazanym lekiem hipo-

tensyjnym?" zamieniane jest przez maszyny wnioskująca na zestaw pytań "Czy X jest wskazanym lekiem hipotensyjnym?", gdzie pod X podstawiane są kolejno leki stosowane w leczeniu nadciśnienia (ale nie inne). Osiągnięcie każdego z celów podstawowych wymaga uprzedniego zrealizowania celów pośrednich. Np. w przypadku nadciśnienia tętniczego należy uprzednio ustalić, czy leczenie farmakologiczne jest w ogóle konieczne, a jeśli tak, to czy ma ono być prowadzone jednym lekiem, czy też kilkoma lekami równocześnie. Ustalenie konieczności i sposobu leczenia wymaga ustalenia dalszych faktów, np. rozstrzygnięcia, czy pacjent jest otyły, do czego potrzebne jest wyznaczenie (obliczenie) jego wagi należnej itd. Wykaz tego rodzaju celów pośrednich i zasady ich osiągania wynikają w prosty sposób z analizy przedstawionej w pracy Wójcik-Jawień [23] i powtarzanie ich tutaj wydaje się niepotrzebne.

Dla każdego leku X, który wystąpił w takim jak postawione wyżej pytaniu, celem pośrednim jest ustalenie przydatności leku X w ogóle, a więc udzielenie odpowiedzi na pytanie "Czy X jest wskazanym lekiem?". Jednakże, w przeciwieństwie do opisanej na wstępie sytuacji, analizowane będą jedynie leki istotne w danym przypadku.

## 4. REALIZACJA PROGRAMU

Zgodnie z przedstawionymi założeniami projektowymi zrealizowano program pozwalający na projektowanie, tworzenie i eksploatację złożonej relacyjnej bazy danych oraz systemu doradczego. Programowi nadano roboczą nazwę THESIS (Tool for building Hard-disk - resident Expert Systems and Information Systems). Niniejsza część pracy omawia strukturę programu i szereg rozwiązań, jakie przyjęto dla realizacji projektu.

### 4.1. Wybór języka programowania.

Cały omawiany system zrealizowany został w języku PASCAL [63-65]. Procedury realizujące dialog z użytkownikiem, nadzorujące pracę całości systemu i przede wszystkim procedury tworzące SZBD zostały w znacznej części zrealizowane na minikomputerze "MERA-60" przy wykorzystaniu kompilatora OMSI PASCAL 1 [66]. Procedury te zostały napisane przy zachowaniu maksymalnej możliwej zgodności ze standardem języka, aby zminimalizować nakład pracy przy przenoszeniu systemu na inny komputer. W związku z uciążliwością pracy na minikomputerze "MERA-60" (hałas, a przede wszystkim częste awarie sprzętu), a także biorąc pod uwagę znaczną popularność komputerów klasy IBM PC-XT/AT, zdecydowano przenieść program na taki mikrokomputer. Po dopasowaniu do wymogów translatora program uruchomiono, a następnie rozbudowano przy użyciu procesora "TURBO-PASCAL" [67, 68] (początkowo wersja 3, następnie 4). Obecnie zostaną przedstawione obiektywne argumenty motywujące wybór tego języka, choć głównym źródłem wielu decyzji związanych z realizacją systemu były środki sprzętowe i programowe, którymi dysponuje Zakład Chemii Fizycznej AM ("MERA-60" i dostępne u producenta oprogramowanie tego minikomputera).

Wybór języka PASCAL uzasadniony był następującymi faktami:

1° PASCAL jest językiem strukturalnym, oferującym programiście szereg środków ułatwiających posługiwanie się obiektami o złożonej strukturze logicznej [59]. Dzięki narzuceniu odpowiedniej dyscypliny "wymusza" staranne programowanie i pozwala na uniknięcie (a przynajmniej wczesne wykrycie) wielu błędów.

2° Translatory tego języka opracowano dla wszystkich liczących się modeli komputerów (a nawet dla domowych komputerów-zabawek), co stwarza możliwość przenoszenia programu na inne komputery (z czego zresztą skorzystano w niniejszej pracy).

3° Kod tworzony przez translatory PASCAL-a z reguły odznacza się dobrą efektywnością, zwłaszcza w zakresie programów nienumerycznych.

4° Autor posiada kilkuletnią praktykę w programowaniu w tym języku na różnych komputerach.

Jako alternatywy dla PASCAL-a rozważono następujące języki lub systemy programowania:

- język C [69]
- LISP [70] i PROLOG [41, 42]
- dBASE III [26, 71]
- EXSYS [72]

Język C zapewnia (pod warunkiem umiejętnego programowania) praktycznie stuprocentową przenośność programów, pozwalając równocześnie na bardzo oszczędne kodowanie. Posiada również porównywalne z PASCAL-em możliwości przetwarzania obiektów o złożonej strukturze. Translatory tego języka generują efektywny kod wynikowy. C idealnie nadaje się do tworzenia oprogramowania systemowego. Jednakże, ze względu na nieco większe niż w przypadku programowania w PASCAL-u niebezpieczeństwo popełnienia trudnych do wykrycia błędów (dyscyplina narzucana przez C jest nieco "luźniejsza"), zdecydowano, że do realizacji prototypowego programu wykorzystany zosta-

nie PASCAL.

Istotną częścią tworzonego programu jest system doradczy. Znaczną część istniejących SD tworzono wykorzystując klasyczne dla zagadnień SI języki, takie jak LISP czy PROLOG. Programy zrealizowane w tych językach wymagają jednak na ogół albo wyspecjalizowanego sprzętu [73], albo dużej pamięci operacyjnej. Języki te lepiej nadają się do przetwarzania struktur reprezentowanych w pamięci operacyjnej; środki do przetwarzania informacji zawartej w pamięciach zewnętrznych są w nich raczej ograniczone. Realizacja omawianego systemu na mikrokomputerze klasy IBM XT byłaby więc obciążona groźbą przepełnienia pamięci, narastającą w miarę wzrostu objętości bazy wiedzy i stopnia złożoności programu. Poza tym w skład systemu wchodzi stosunkowo obszerna (nie mieszcząca się w pamięci operacyjnej) baza danych. Stosowanie języków SI w ich obecnej postaci nie wydaje się odpowiednie do konstrukcji większych SZBD. Należy jednak nadmienić, że niewygodny tradycyjnej notacji LISP-owej [70] nie są tu istotnym argumentem - np. autor opracował w 1981 pascalopodobny język programowania i translator tłumaczący konstrukcje tego języka na LISP [74]. Pomimo nabytego przy tym pewnego doświadczenia w programowaniu w LISP-ie, w świetle przedstawionych wyżej mankamentów nie zdecydowano się na użycie LISP-u ani PROLOG-u. Warto tu zwrócić uwagę na zarysowującą się obecnie tendencję do odchodzenia od stosowania języków SI przy realizacji SD [75].

Program dBASE jest typowym SZBD, wyposażonym we własny, strukturalny język pozwalający budować programy wykorzystujące bazy danych. Oferowane przez ten język środki nie związane z obsługą bazy danych są jednak ograniczone i nie wydają się wystarczające do tworzenia bardziej złożonych zastosowań. Główną słabością systemu dBASE, która go ostatecznie wyeliminowała, jest brak odpowiednich środków do efektywnego przetwarzania danych o zmiennej długości, takich jak informacja tekstowa (ograniczenia nałożone na tzw. pola

"MEMO" są zbyt rygorystyczne - np. brak możliwości wyszukiwania wg zawartości tych pól). Wprowadzony ostatnio na rynek program dBASE IV wady tej już nie posiada.

Program EXSYS jest programem umożliwiającym tworzenie i eksploatację systemu doradczego z regułową bazą wiedzy. Nie posiada on jednak gotowych środków do komunikacji z bazami danych, co utrudnia realizację informacyjnej bazy danych, jak również bazy danych o pacjentach. Również wielkość bazy wiedzy jest ograniczona pojemnością pamięci operacyjnej, tak że podejmowanie budowy większego systemu jest ryzykowne. Pewną wadą jest również konieczność konwersacji systemu z użytkownikiem w języku angielskim. Mimo niewątpliwych walorów tego programu, wobec wymienionych braków nie zdecydowano się na jego wykorzystanie.



#### 4.2. Struktury danych.

THESIS operuje obiektami o złożonej strukturze logicznej, takimi jak schemat bazy danych, jej organizacja fizyczna, stwierdzenia, reguły wnioskowania, wyrażenia arytmetyczne i wiele innych. Obiekty te muszą być odpowiednio reprezentowane w pamięci operacyjnej (lub zewnętrznej). W niniejszym paragrafie przedstawione zostaną najważniejsze struktury danych zastosowane w programie. Zasady ich wykorzystania omawiane są w punkcie 4.3.

##### 4.2.1. Słowniki i identyfikatory.

Słowniki służą do przechowywania informacji o identyfikatorach. Identyfikatorami są nazwy relacji, atrybutów, nazwy i składniki typów wyliczeniowych oraz zmienne lokalne. Każdy słownik jest zorganizowany w postaci tablicy rozproszonej (ang. *hash table*) [61]. Strukturę słownika i identyfikatorów ilustruje ryc. 4.1. Każda relacja ma swój słownik, w którym przechowywane są identyfikatory jej atrybutów. Poza tym istnieje jeden słownik globalny. Znajdują się w nim nazwy relacji oraz nazwy i składniki typów wyliczeniowych.

Opis identyfikatora  
(element listy identyfikatorów)

SŁOWNIK:

wskaźnik do listy identyfikatorów	<1>
.....	<2>
.....	
wskaźnik do listy identyfikatorów	<n>

nazwa identyfikatora
rodzaj identyfikatora (zmienna, atrybut BD itp.)
typ wartości
aktualna wartość
wskaźnik do następnego identyfikatora na liście
dane zależne od semantyki identyfikatora

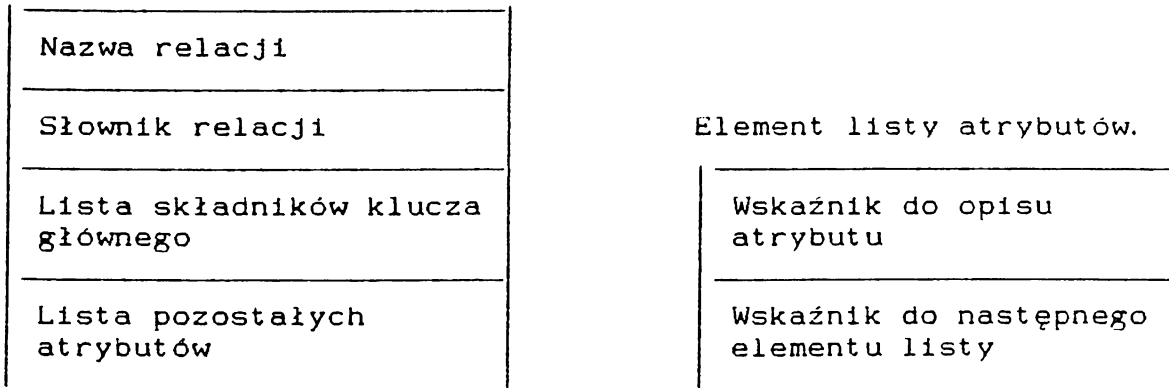
Ryc. 4.1. Struktura słownika i identyfikatorów.

#### 4.2.2. Reprezentacja struktury bazy danych.

Struktura BD jest reprezentowana w postaci tablicy opisów relacji. Deskryptor relacji zawiera informacje o jej logicznej (ryc. 4.2) i fizycznej (ryc. 4.3) organizacji.

Schemat logiczny tworzą dwie listy atrybutów: lista składników klucza głównego i lista pozostałych atrybutów relacji. Atrybuty są na tych listach reprezentowane w postaci wskaźników do ich opisów, przechowywanych w lokalnym słowniku relacji. Wskaźnik do tego słownika umieszczony jest również w deskrypcji relacji.

#### DESKRYPTOR RELACJI.



Ryc. 4.2. Reprezentacja struktury logicznej relacji.

Informacje o fizycznej strukturze relacji zawarte są głównie na tzw. liście dostępu. Każdemu atrybutowi wchodzącemu w skład klucza odpowiada jedno ogniwo tej listy, zawierające informacje niezbędne przy realizacji dostępu wg klucza złożonego. Dostęp zazwyczaj realizowany jest w kilku etapach i dlatego przedstawienie drogi dostępu w postaci listy wydaje się odpowiednie. Na informacje te składają się m.in. rozmiar rekordu, numer pliku dyskowego (dane odnośnie plików bazy danych są przechowywane w oddzielnej tablicy), organizacja tego pliku (sekwencyjna, rozproszona itp.). Lista dostępu posiada wskaźniki do trzech list zawierających opisy pól re-

kordu. Pierwsza z tych list jest jednoelementowa i opisuje bieżący składnik klucza w rekordzie. Druga występuje tylko dla ostatniego składnika klucza i obejmuje wszystkie atrybuty spoza klucza głównego. Trzecia opisuje pola rekordu nie powiązane z żadnym atrybutem, a przeznaczone do przechowywania danych ściśle związanych ze stosowaną metodą dostępu (np. wskaźniki do rekordów nadmiarowych w organizacji rozproszonej).

Na liście dostępu znajdują się również dane zmieniające się dynamicznie podczas pracy programu, np. absolutna pozycja aktualnego rekordu.

Element listy dostępu:

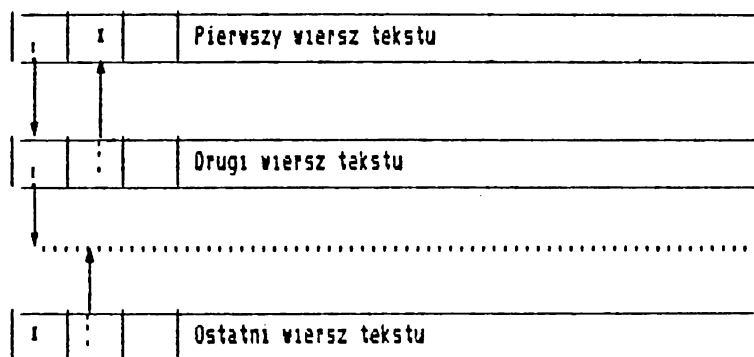
Jednoelementowa lista zawierająca opis pola odpowiadającego kluczowi
Lista opisów pól rekordu odpowiadających pozostałym atrybutom
Lista opisów pól "technicznych"
Wskaźnik do następnego elementu listy dostępu
Organizacja pliku
Numer pliku dyskowego
Aktualna pozycja pliku
Długość rekordu
Zależne od organizacji pliku dane potrzebne do zrealizowania dostępu do rekordu

Ryc. 4.3. Element tzw. listy dostępu będącej opisem struktury fizycznej relacji.

#### 4.2.3. Informacja tekstowa.

Większość atrybutów bazy danych w przewidywanym zastosowaniu ma

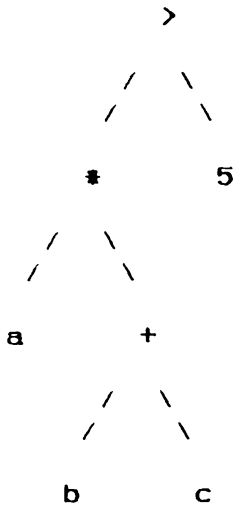
charakter tekstowy. Informacja ta, określana jako "tekst dowolnej długości", w czasie jej przetwarzania (wyświetlania, redagowania itp.) jest reprezentowana w pamięci operacyjnej w postaci listy wierszy (ryc. 4.4). Każdy element tej listy, oprócz obrazu wiersza (tablica znakowa), posiada wskaźniki do poprzedniego i następnego wiersza oraz położenie ostatniego znaku w wierszu. Ta sama struktura używana jest do przechowywania sekwencyjnych ciągów bajtów o innym niż tekstowe znaczeniu (jest to wykorzystywane zwłaszcza w reprezentacji wiedzy w postaci reguł wnioskowania).



Ryc. 4.4. Reprezentacja informacji tekstowej w pamięci operacyjnej.

#### 4.2.4. Wyrażenia arytmetyczne i zapytania do bazy danych.

Przy przeszukiwaniu bazy danych lub podczas wnioskowania zachodzi potrzeba obliczania wyrażeń arytmetyczno-logicznych. Aby interpretacja tych wyrażeń była możliwa, muszą one być w odpowiedni sposób reprezentowane w pamięci. W omawianym rozwiązaniu wybrano często stosowaną do tego celu strukturę drzewiastą, przedstawioną na ryc. 4.5. Ponieważ jednak elementy reguł przechowywane są w bazie danych w postaci sekwencyjnej (zob. 4.2.5), dlatego wyrażenia wchodzące w skład reguł zapisywane są w notacji polskiej<sup>1</sup> i przed obliczaniem przekształcane do struktury drzewiastej.



Ryc. 4.5. Reprezentacja wyrażenia:  $a*(b+c) > 5$  w postaci drzewa.

#### 4.2.5. Reprezentacja wiedzy.

Zgodnie z przyjętymi założeniami, wiedza reprezentowana jest przy użyciu reguł wnioskowania. Reguły te mają postać:

```
JEŻELI
  < 1-szy element przesłanki >
ORAZ
  < 2-gi element przesłanki >
.....
ORAZ
  < n-ty element przesłanki >
TO
  < 1-szy wniosek >
ORAZ
  < 2-gi wniosek >
.....
ORAZ
  < m-ty wniosek >
PRZECIWNIE
  < 1-szy przeciwny wniosek >
ORAZ
  < 2-gi przeciwny wniosek >
.....
ORAZ
  < p-ty przeciwny wniosek >
```

Elementami przesłanek mogą być:

- wyrażenia arytmetyczno-logiczne
- stwierdzenia

Wnioski mogą obejmować:

- stwierdzenia
- komunikaty
- obliczenia arytmetyczne

Z każdym stwierdzeniem występującym we wniosku (prostym lub przeciwnym) stowarzyszony jest stopień pewności tego wniosku.

Baza danych służąca reprezentacji wiedzy i wnioskowaniu składa się z trzech relacji tworzących schemat przedstawiony na ryc. 4.6.

OKREŚLENIA:

Określenia	Twierdząca postać orzeczenia	Przecząca postać orzeczenia	Lista stwierdzeń dla przesłanki	Lista stwierdzeń dla wniosku	Lista stwierdzeń dla przeciwnego wniosku
------------	------------------------------	-----------------------------	---------------------------------	------------------------------	--

STWIERDZENIA:

Podmiot	Określenie	Lista reguł dla przesłanki	Lista reguł dla wniosku	Lista reguł dla przeciwnego wniosku	MB (miara zaufania)	MD (miara wątpliwości)
---------	------------	----------------------------	-------------------------	-------------------------------------	---------------------	------------------------

Status stwierdzenia	Wskaźnik do pozycji na stosie ocenianych stwierdzeń
---------------------	---

REGUŁY:

Numer reguły	Przesłanka	Wniosek (prosty)	Wniosek przeciwny	Status reguły	Komentarz do wniosku	Komentarz do przeciwnego wniosku
--------------	------------	------------------	-------------------	---------------	----------------------	----------------------------------

Wskaźnik do pozycji na stosie aktywnych reguł
---

Ryc. 4.6 Schemat relacji wykorzystywanych do regułowej reprezentacji wiedzy.

Projektując organizację fizyczną przyjęto dla reguł organizację absolutną (numer reguły określa pozycję fizyczną), dla określeń zastosowano kodowanie rozpraszające. Relacja STWIERDZENIA ma dwa klucze: dostęp według pierwszego (PODMIOT) realizowany jest techniką rozpraszającą, drugi klucz (OKREŚLENIE) identyfikowany jest sekwencyjnie. Zastosowanie techniki rozpraszającej jest tu szczególnie korzystne, ze względu na charakterystyczną dla niej niezależność czasu dostępu od rozmiaru bazy danych.

Atrybuty takie jak 'Lista stwierdzeń...', 'Lista reguł...', a także 'Przesłanka', 'Wniosek' oraz 'Wniosek przeciwny' zapisywane są w postaci ciągu bajtów. Na uwagę zasługuje sposób reprezentowania elementów reguły. Format zapisu elementów przesłanki (lub wniosku) przedstawia ryc. 4.7. Pierwszy bajt pozwala określić rodzaj elementu. Jeśli jego najstarszy bit jest wyzerowany, zawartość tego i trzech następnych bajtów identyfikuje stwierdzenie. Wtedy szósty bit pierwszego bajtu wskazuje, czy w stwierdzeniu występuje twierdząca, czy przecząca postać orzeczenia. Gdy najstarszy bit pierwszego bajtu jest ustawiony, lecz wartość bajtu jest mniejsza od 240, jest ona interpretowana jako kod operatora. Wtedy ten i następne bajty stanowią reprezentację wyrażenia arytmetyczno-logicznego w prostej notacji polskiej. Gdy wreszcie wartość numeryczna bajtu jest zawarta w zakresie 240-255, jest ona traktowana jako prefiks, zapowiadający - w zależności od swej wartości - komunikat, identyfikator atrybutu, stałą tekstową, numeryczną itp. Kolejne elementy przesłanki następują bezpośrednio po sobie (wiadomo, że muszą one być połączone spójnikiem ORAZ - nie trzeba więc tego spójnika zapisywać). Koniec przesłanki sygnalizuje bajt o wartości 255. Reprezentacja wniosku (czy wniosku przeciwnego) jest podobna; główna różnica polega na tym, że zapis stwierdzenia uzupełniany jest podaniem stopnia pewności reguły w odniesieniu do konkluzji wyrażonej przez to stwierdzenie.

DESKRYPTOR STWIERDZENIA

0	x	bardziej znacząca część numeru określenia
mniej znaczący bajt numeru podmiotu		
bardziej znaczący bajt numeru podmiotu		
mniej znaczący bajt numeru określenia		
Zakodowany w dwu bajtach stopień pewności wniosku		

$$x = \begin{cases} 0, & \text{gdy twierdząca postać stwierdzenia} \\ 1, & \text{gdy przecząca postać stwierdzenia} \end{cases}$$

(tylko dla stwierdzeń występujących we wnioskach)

OPERATOR

1	kod operatora (< 112)
---	-----------------------

KOMUNIKAT

2 4 9
pierwszy znak komunikatu
.....
ostatni znak komunikatu
2 5 5

IDENTYFIKATOR ATRYBUTU

2 5 2
pierwszy znak nazwy relacji
.....
ostatni znak nazwy relacji
2 5 5
pierwszy znak identyfikatora atrybutu
.....
ostatni znak identyfikatora atrybutu
2 5 5

Ryc. 4.7. Format zapisu niektórych elementów przesłanek i wniosków.



#### 4.3. Elementy systemu.

THESIS jest zintegrowanym programem umożliwiającym projektowanie, tworzenie i eksploatację relacyjnej bazy danych oraz konstrukcję i eksploatację systemu doradczego opartego o wiedzę wyrażoną w postaci reguł wnioskowania. Głównymi elementami programu są: podstawowe moduły SZBD, moduły wprowadzania, prezentacji i wyszukiwania informacji, moduły stanowiące podstawę dla konstrukcji systemu doradczego (ang. *expert system shell*) i moduły eksploatacji tego systemu, translator i interpreter wyrażeń arytmetyczno-logicznych oraz moduły komunikacji z użytkownikiem (ang. *user interface*). Elementy jednego rodzaju intensywnie używają innych; np. maszyna wnioskująca systemu doradczego korzysta stale z SZBD, wszystkie moduły korzystają ze środków zapewniających komunikację z użytkownikiem, te ostatnie zaś znowu używają SZBD itd.

##### 4.3.1. Moduły SZBD.

SZBD składa się z procedur umożliwiających projektowanie logicznej i fizycznej struktury danych, procedur wykonujących podstawowe operacje przetwarzania danych (wyszukiwanie, wstawianie, odczyt, zapis (aktualizacja) i kasowanie rekordów) oraz modułów wprowadzania, prezentacji i wyszukiwania informacji.

##### 4.3.1.1. Edytory struktury danych.

Edytor logicznej struktury danych umożliwia wyszczególnienie relacji tworzących bazę danych, określenie atrybutów relacji i wskazanie, które z nich tworzą klucz główny. Edytor reprezentacji fizycznej pozwala dla każdego z atrybutów określić sposób kodowania, a w razie potrzeby długość pola rekordu. Dla każdej relacji określa się organizację fizyczną pliku, tj. strategię lokalizacji rekordu wg poszczególnych składników klucza. Wyboru sposobu kodowania danych czy metody dostępu do rekordu dokonuje się przy pomocy

odpowiedniego "menu" (realizowanego przy użyciu procedur komunikacji z użytkownikiem, omawianych w punkcie 4.3.5). Efektem wykonania tych procedur jest utworzenie wewnętrznej reprezentacji logicznej i fizycznej struktury relacji tworzących bazę danych. Reprezentację tę omówiono w punkcie 4.2.2.

#### 4.3.1.2. Procedury podstawowe.

W tej grupie znajdują się procedury lokalizacji rekordu na podstawie znajomości wartości atrybutów klucza, procedury pobierania i zapisywania zawartości wybranego pola uprzednio zlokalizowanego rekordu, a także, przeznaczone do przeszukiwania bazy danych, procedury ustawienia pierwszego rekordu relacji i ustawienia następnego rekordu po ostatnio ustawionym. Wszystkie te procedury korzystają z reprezentowanego w pamięci modelu fizycznej organizacji danych.

Procedury lokalizacji, w zależności od wybranego trybu pracy, mogą wyszukiwać rekord identyfikowany kluczem (tryb FIND), tworzyć taki rekord - o ile już nie istniał (tryb CREATE) i kasować go - o ile istnieje (tryb DELETE). Każdorazowo procedury lokalizacji przekazują informację o tym, czy konkretna krotka istniała p r z e d rozpoczęciem operacji.

#### 4.3.1.3. Moduły wprowadzania, prezentacji i wyszukiwania informacji.

W skład tych modułów wchodzi procedury planowania rozmieszczenia informacji na ekranie, edytor danych (pozwalający na wypełnianie, wyświetlenie i ewentualne modyfikacje poszczególnych pól rekordu), edytor zapytań do bazy danych, interpreter zapytań i procedury prezentacji wyników wyszukiwania.

Procedury planowania rozmieszczenia informacji na ekranie pozwalają określić rozmieszczenie pól rekordu wyświetlanych na ekranie, przy-

pisać im tytuły (mogą one, ale nie muszą pokrywać się z nazwą odpowiedniego atrybutu) i określić wartości domyślne (ang. *default*) atrybutów (używane w czasie redagowania nowych krotek). Te same procedury służą również do planowania wydruku. Wynikiem ich pracy jest utworzenie w pamięci struktury określającej format prezentacji danych. Struktura taka jest następnie zapamiętywana na dysku przy użyciu procedur omawianych w punkcie 4.3.4.

Edytor danych wyświetla na ekranie zaprojektowany uprzednio formularz i oczekuje na wpisanie atrybutów tworzących klucz główny. Po ich wprowadzeniu następuje zgodne z fizyczną strukturą relacji zakodowanie składników klucza i odwołanie do podstawowych procedur SZBD w celu sprawdzenia, czy odpowiednia krotka znajduje się w bazie danych. Gdy tak jest, dokonuje się pobrania wartości pozostałych atrybutów i ich rozkodowania. W przeciwnym wypadku atrybutom nadawane są wartości domyślne, określone w czasie projektowania formularza. Następnie edytor umożliwia użytkownikowi przeglądanie i redakcję danych. Po zakończeniu redakcji, stosownie do życzenia użytkownika, wskazana krotka wpisywana jest do bazy danych (być może aktualizując już istniejącą, identyfikowaną tym samym kluczem) lub jest ona usuwana. Użytkownik może również zaniechać wprowadzania zmian w bazie danych, zadowolając się przeglądnięciem informacji. Prawo modyfikacji pewnych relacji może być zastrzeżone tylko dla administratora bazy danych (uprawnienia użytkownika system rozpoznaje na podstawie identyfikatora podawanego na żądanie programu).

Edytor zapytań funkcjonuje podobnie do edytora danych (w istocie korzysta on z tych samych procedur; odpowiedni parametr wskazuje aktualny rodzaj pracy). Dla każdej wybranej przez użytkownika relacji bazy danych wyświetlany jest odpowiedni formularz. Użytkownik dla wybranych pól określa warunki, jakie mają one spełniać w poszukiwanych krotkach. Określenie warunku polega na podaniu re-

lacji (arytmetycznej, logicznej itp.), jaka ma zachodzić pomiędzy wartością rozważanego atrybutu, a wartością wpisaną na formularzu. Na przykład, chcąc wyszukać te leki z I grupy wg Williama, które są przeciwwskazane w chorobie wrzodowej, należy w polu GRUPA FARMAKOLOGICZNA wpisać "I grupa wg Williama" i wybrać relację 'Równe', a w polu PRZECIWWSKAZANIA wpisać "choroba wrzodowa" i wybrać relację 'Zawiera'. Tak zredagowany warunek jest koniunkcją warunków składowych. Kolejnym etapem redagowania zapytania jest określenie, które atrybuty w wyszukanych krotkach mają być wyświetlone (badź wydrukowane). Wyboru tych atrybutów dokonuje się przy pomocy odpowiedniego "menu", wyświetlającego atrybuty danej relacji BD. Etap ten może być pominięty, jeśli użytkownik zgodzi się na standardowy zestaw atrybutów (przygotowany przez administratora BD). Gdy pytanie dotyczy atrybutów jednej tylko relacji bazy danych, redagowanie pytania na tym się kończy. Gdy jednak warunek wyszukiwania lub lista potrzebnych atrybutów obejmują więcej niż jedną relację, zachodzi potrzeba określenia powiązań. Można skorzystać z powiązania standardowego lub przejść do redagowania powiązania. Standardowe powiązanie określa administrator bazy danych; w przedstawionym zastosowaniu poszczególne relacje powiązane są atrybutami o nazwie NAZWA MIĘDZYNARODOWA. Redagowanie powiązania przebiega analogicznie jak redagowanie warunku, ale w poszczególnych polach można teraz wpisywać jedynie nazwy zmiennych wiążących. Wyboru zmiennej dokonuje się przy pomocy "menu"; możliwe nazwy zmiennych to -A-, -B-, itd. Na przykład, aby określić standardowe powiązanie dla bazy danych o leku, należy dla każdej relacji bazy danych wpisać w polu NAZWA MIĘDZYNARODOWA tę samą zmienną (powiedzmy -A-) i wybrać relację 'Równe'. Po określeniu powiązania redakcja zapytania (a tym samym praca edytora zapytań) zostaje zakończona. Zredagowane zapytania i powiązania są pamiętane w postaci drzewiastej struktury, analogicznej do tej jaką tworzy translator wyrażeń arytmety-

czno-logicznych (por. ryc. 4.5). Użytkownik może zapisać na dysku określone przez siebie powiązania i zestawy atrybutów i w przyszłości odwoływać się do nich.

Procedury wyszukiwawcze przeszukują bazę danych interpretując drzewo zapytania. Gdy zostaje znaleziony rekord spełniający wymagane warunki, procedura wyszukiwawcza przekazuje działanie procedurom prezentacji wyników, które wyświetlają lub drukują wymagane przez użytkownika pola. Procedury interpretacji drzewa zapytania starają się zminimalizować nakład pracy przy przeszukiwaniu bazy danych; np. wykorzystują one fakt, że występujący w pytaniu atrybut jest kluczem (składnikiem klucza) relacji. Do wyszukiwania wzorca w tekście zastosowano efektywny algorytm Boyera-Moore'a [76]. Obecnie nie dokonuje się wstępnej optymalizacji drzewa zapytań [25, 77], w przypadku rozbudowy systemu odpowiednie procedury powinny zostać dołączone. Ze względu na występowanie powiązań, procedury wyszukiwawcze wyposażone są w mechanizm uzgadniania (unifikacji) atrybutów i zmiennych wiążących.

#### 4.3.1.4. Procedury zapisu i odczytu struktur bazy danych.

Zadaniem procedur z tej grupy jest zapisywanie w pamięci dyskowej danych o strukturze i aktualnym stanie bazy danych, a także odczyt tych danych. Procedury odczytu uruchamiane są automatycznie w czasie rozpoczynania pracy z bazą danych; jeśli w wyniku działalności administratora bazy danych nastąpiły jakiegokolwiek zmiany, przed zakończeniem pracy programu dokonany zostaje zapis odpowiednich danych. W tej grupie są również procedury zapisujące i odczytujące plan formularzy służących do wprowadzania i prezentacji danych.

#### 4.3.2. Translator wyrażeń arytmetyczno-logicznych.

Procedury translatora wykorzystywane są przez edytor reguł. Pozwalają one umieszczać w treści reguł sprawdzanie warunków arytmetyczno-logicznych lub wykonywanie odpowiednich obliczeń. Translator składa się z analizatora leksykalnego, parsera (analizatora składni) i makrogeneratora (wykonującego analizę semantyczną) [78-82].

Analizator leksykalny pobiera znaki ze wskazanego źródła (kanału), identyfikując podstawowe jednostki leksykalne. Analizator rozpoznaje liczby, identyfikatory (nazwy stałych, zmiennych, relacji i atrybutów) i operatory. Opis rozpoznawanych jednostek leksykalnych w notacji EBNF<sup>2</sup> jest następujący:

```
<jednostka leksykalna> ::= <liczba> | <identyfikator> | <operator>
<liczba> ::= <część całkowita> { . <część ułamkowa> }
<część całkowita> ::= <ciąg cyfr>
<część ułamkowa> ::= <ciąg cyfr>
<ciąg cyfr> ::= <cyfra> | <cyfra> <ciąg cyfr>
<cyfra> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
<identyfikator> ::= <litera> | <litera> <ciąg znaków>
<ciąg znaków> ::= <znak> | <znak> <ciąg znaków>
<znak> ::= <litera> | <cyfra> | _
<litera> ::= <mała litera> | <wielka litera>
<operator> ::= + | - | * | / | = | > | < | >= | <= | # | $ | & |
           ! | . | := | ;
```

Znaczenie symboli <mała litera> i <wielka litera> jest oczywiste.

Gdy jednostka leksykalna zostaje rozpoznana jako identyfikator, następuje sprawdzenie, czy występuje on w aktualnym słowniku. Dokonują tego procedury zarządzające słownikiem. W słowniku zawarte są niezbędne informacje o semantyce identyfikatora. Określenie identyfikatorów następuje podczas projektowania relacji bazy danych. W przypadku napotkania nieznanego identyfikatora analizator leksykalny sygnalizuje błąd. W czasie pracy analizator leksykalny korzysta z aktualnego słownika, którym zwykle jest słownik globalny. Jeśli jednak procedura nadzorująca analizę leksykalną napotka nazwę relacji, a po niej rozpozna operator '.' (kropkę), to następuje chwilowa zmiana aktualnego słownika na słownik związany z tą rela-

cja. Po wprowadzeniu następnego obiektu przywracany jest dotychczasowy słownik. W tej sytuacji ów ostatni obiekt, (o ile jest poprawny) zostaje przekazany jako końcowy rezultat analizy leksykalnej.

Parser (analizator składni) pobiera kolejne jednostki od analizatora leksykalnego, sprawdza poprawność utworzonego z nich wyrażenia i tworzy drzewo rozbioru. Gramatyka, której podlegają wyrażenia jest gramatyką operatorową z pierwszeństwami [78,80]. Rozbiór realizowany jest metodą zstępującą (ang. *top-down*) [82]. Tabela 4.1 przedstawia operatory uszeregowane według rosnących priorytetów.

Tabela 4.1. Operatory pogrupowane wg malejących priorytetów.

Symbol operatora	Znaczenie
.	dostęp do atrybutu relacji
*	mnożenie
/	dzielenie
\$	należy do zbioru
~	nie należy do zbioru
+	dodawanie
-	odejmowanie
=	równe
<	mniejsze
<=	mniejsze lub równe
>	większe
>=	większe lub równe
#	nierówne
&	koniunkcja
!	alternatywa
:=	przypisanie wartości

Np. wyrażenie:

$a + b > 5 \ \& \ c * d \geq 2 \ ! \ 3 * 4 \ \$ \ f * g$

należy rozumieć jako:

$((a + b) > 5) \ \& \ ((c * d) \geq 2) \ ! \ ((3 * 4) \$ (f * g))$

Przykładową strukturę drzewa rozbioru przedstawiono na ryc. 4.5.

Odpowiednie drzewo rozbioru jest również rezultatem pracy procedur

redagujących zapytania do bazy danych. Dzięki przejściu konstrukcji tego drzewa przez edytor zapytań unika się niebezpieczeństwa częstych omyłek, a użytkownik nie musi uczyć się zasad poprawnego konstruowania wyrażeń.

Makrogenerator dokonuje wstępnej analizy semantycznej utworzonego przez parser drzewa rozbioru. W wyrażeniach mogą występować operatory polimorficzne, tj. takie, których znaczenie zależy od typu operandów. Np  $a > b$ , gdzie  $a$  i  $b$  są liczbami, ma sens porównania liczb względem ich naturalnego uporządkowania, natomiast gdy  $a$  i  $b$  są łańcuchami znaków, ten sam zapis oznacza porównanie względem porządku leksykograficznego. Gdy zaś  $a$  jest liczbą natomiast  $b$  - łańcuchem znaków, zapis  $a > b$  jest po prostu pozbawiony sensu (a więc błędny). Praca makrogeneratora polega na zmianie struktury drzewa, stosownie do właściwego znaczenia wyrażenia. Makrogenerator wykorzystywany jest zarówno przez maszynę wnioskującą systemu doradczego, jak i przez procedury wyszukiwawcze do przetwarzania drzewa zapytania.

Interpreter oblicza wartości wyrażeń. Główną częścią interpretera jest procedura realizująca rekursywny algorytm: Gdy węzeł drzewa rozbioru zawiera atrybut, stałą lub zmienną - pobierana jest ich wartość. Gdy węzeł zawiera operator, przy pomocy tej samej procedury oblicza się wartości jego argumentów (gałęzi drzewa), a następnie wykonuje czynności (obliczenia itp.) określone przez operator. Jeśli w czasie wykorzystywania interpretera przez maszynę wnioskującą napotkany zostanie atrybut o nieokreślonej wartości, interpreter wywoła procedury redagowania danych i zażąda od użytkownika systemu doradczego dostarczenia brakujących informacji.



#### 4.3.3. Moduł dla tworzenia systemu doradczego.

Moduł ten posiada edytor bazy wiedzy współpracujący z procedurami przekształcającymi wiedzę do odpowiedniej reprezentacji wewnętrznej.

##### 4.3.3.1. Edytor reguł.

Edytor umożliwia tworzenie nowych, modyfikację istniejących i usuwanie niepotrzebnych reguł. Jest on zintegrowany z resztą systemu i dzięki temu zapewnia bieżącą kontrolę poprawności reguł, wspomagając proces pozyskiwania wiedzy.

Jest to edytor ekranowy, zrealizowany z wykorzystaniem techniki okienek i rozwijanego "menu". Do przedstawienia reguły używane są spójniki (JEŻELI, ORAZ itd.) i jednostki (stwierdzenia, komunikaty i in.). Każda jednostka prezentowana jest w jednym wierszu na ekranie. W czasie redagowania jednostki edytor pyta o jej rodzaj (STWIERDZENIE, WYRAŻENIE, KOMUNIKAT itp.), a po zredagowaniu o spójnik, którego ma użyć.

Stwierdzenie redagowane jest w jednym wierszu ekranu, podzielonym na trzy pola (okienka) odpowiadające części podmiotu, orzeczeniu i określeniu (ryc. 4.8). Po wpisaniu podmiotu przez użytkownika, edytor sprawdza, czy podmiot ten jest znany (tj. czy istnieje w bazie danych). Jeśli nie, upewnia się, pytając użytkownika, czy nie popełnił omyłki.

PRZEWLEKŁA NIETYDOLNOŚĆ KRAŻENIA	JEST	STWIERDZONYM SCHORZENIEM
Pole części podmiotu	Pole orzeczenia	Pole określenia

Ryc. 4.8. Podział wiersza ekranu przeznaczonego do redagowania stwierdzenia.

Jeśli podmiot jest znany, edytor wyświetla jako propozycję określenie, z którym tego podmiotu użyto po raz pierwszy. Użytkownik może przyjąć tę propozycję lub wpisać inne określenie. Również w przy-

padku wpisania określenia, które nie jest jeszcze znane, edytor żąda rozstrzygnięcia, czy jest to nowe określenie, czy tylko pomyłka w zapisie istniejącego. W razie pojawienia się nowego określenia należy wpisać odpowiednią postać orzeczenia, a także podać brzmienie jej zaprzeczenia. Gdy określenie jest znane, edytor wyświetla orzeczenie, którego postać może być zmieniana odpowiednim klawiszem (↑, ↓ lub odstępy) z twierdzącej na przeczącą (i odwrotnie).

Jeśli redagowane stwierdzenie znajduje się we wniosku (lub wniosku przeciwnym), to edytor żąda określenia stopnia pewności. W tym celu wyświetlany jest specjalny diagram (por. ryc. 5.5). Posługując się odpowiednimi klawiszami (←, →) użytkownik przesuwa ruchomy wskaźnik tak, aby najlepiej odpowiadał jego subiektywnej ocenie pewności rozważanego stwierdzenia. Rozwiązanie to ("podpatrzone" w programie INSIGHT 2) pozwala uniknąć konieczności wyrażania stopnia pewności w postaci liczbowej (np. w procentach). Doświadczenie wykazuje, że taki "analogowy" sposób określania stopnia pewności jest chętniej przyjmowany przez użytkowników i z reguły udzielają oni odpowiedzi po krótszym namyśle.

Wyrażenia arytmetyczno-logiczne redagowane są w okienku zajmującym jeden wiersz ekranu. Można w nim jednak redagować dowolnie długie wyrażenia, choć wyświetlany jest tylko bieżący wiersz. Bezpośrednio po zakończeniu redakcji wyrażenia (tj. po wciśnięciu klawisza Esc) zostaje ono przetłumaczone do wewnętrznej reprezentacji. Jeśli wyrażenie jest niepoprawne (np. użyto nieznanego nazwy atrybutu), to sygnalizowany jest błąd (dźwiękiem i wyświetleniem odpowiedniego komunikatu) i edytor wymusza powrót do redagowania wyrażenia, oczekując na poprawki.

Technika redagowania komunikatów jest analogiczna, z tą oczywiście różnicą, że nie jest analizowana ich poprawność.

Po zredagowaniu treści reguły i wyrażeniu przez użytkownika woli jej zapisania w systemie, wykonywane są następujące czynności:

- 1°. Dla każdego stwierdzenia użytego w regule numer tej reguły odnotowany zostaje na odpowiedniej liście.
- 2°. Zapisywane są wszystkie części reguły (tj. przesłanka, wniosek i przeciwny wniosek) oraz stopnie pewności (dla stwierdzeń użytych we wnioskach)

#### 4.3.4. Moduł eksploatacji systemu doradczego.

Składa się on z edytora zapytania do SD, maszyny wnioskującej, procedur prezentacji i objaśniania wyników.

Edytor zapytań wymaga od użytkownika podania wstępnych danych (w omawianym zastosowaniu jest to identyfikator pacjenta i jego podstawowe dane). W tym celu edytor odwołuje się do edytorów danych z SZBD. Użytkownik nie musi wpisywać danych we wszystkich polach; w szczególności może się ograniczyć tylko do podania klucza (np. identyfikatora pacjenta), gdyż maszyna wnioskująca zapyta o brakujące dane, o ile będą one potrzebne. Po zakończeniu redakcji danych dokonywana jest inicjalizacja zbiorów reguł i stwierdzeń. Polega ona głównie na wpisaniu w polu STATUS każdej reguły i każdego stwierdzenia informacji, że nie były one jeszcze użyte w tym wnioskowaniu.

Następnie użytkownik określa rodzaj pytania. Może to być pytanie typu "CZY" albo pytanie typu "CO". Jeśli pytanie jest typu "CZY", to należy podać stwierdzenie, o które pytamy (np. CZY PROPANOLOL JEST WSKAZANYM LEKIEM?). Redakcję stwierdzenia obsługuje ten sam edytor stwierdzeń, z którego korzysta edytor reguł. W przypadku pytania typu "CO" określamy (również korzystając z edytora stwierdzeń, ale pracującego w nieco innym trybie) orzeczenie wraz z określeniem (podmiotu się nie wpisuje, gdyż jest nim zaimek "CO"), np. CO JEST WSKAZANYM LEKIEM?.

Zgodnie z postawionym przez użytkownika pytaniem, edytor konstruuje stos stwierdzeń, które stanowią przedmiot pytania. W przypadku py-

tania typu "CZY" stos zawiera pojedyncze stwierdzenie wyrażone w pytaniu. Gdy pytanie jest typu "CO", z relacji OKREŚLENIA odczytuje się listę stwierdzeń zawierających określenie podane w pytaniu i wszystkie stwierdzenia z tej listy umieszczane są na tzw. stosie badanych stwierdzeń. Następnie sterowanie zostaje przekazane do maszyny wnioskującej, której zadaniem jest ustalenie prawdziwości wszystkich (o ile to możliwe) stwierdzeń zawartych na stosie.

Maszynę wnioskującą tworzy zbiór procedur wybierających właściwą regułę, oceniających prawdziwość przesłanek reguły i wyciągających odpowiednie wnioski, zgodnie ze strategią wnioskowania wstecz.

Praca maszyny wnioskującej polega na powtarzaniu wywołań procedury oceny stwierdzeń, przekazując jej jako parametr stwierdzenie znajdujące się na szczycie stosu. Wywoływanie tej procedury jest powtarzane, aż do opróżnienia stosu. Zadaniem procedury oceny stwierdzeń jest ustalenie prawdziwości stwierdzenia określonego przez parametr. W tym celu (po sprawdzeniu, czy stwierdzenie nie było już oceniane, bo wtedy ponowna ocena jest niepotrzebna!) z relacji STWIERDZENIA pobierana jest lista reguł, w których konkluzji występuje rozważane stwierdzenie. Dla każdej z tych reguł wywoływana jest procedura oceny przesłanek reguły. Jeśli żadna z rozważanych reguł nie umożliwi oceny stwierdzenia, zostaje ono przeniesione ze stosu badanych stwierdzeń na stos stwierdzeń niemożliwych do ustalenia. W przeciwnym wypadku stwierdzenie zostaje przeniesione na stos stwierdzeń ustalonych. W każdej sytuacji stwierdzenie znika ze stosu badanych stwierdzeń.

Procedura oceny przesłanek reguły sprawdza, czy reguła była już oceniana i w takim wypadku natychmiast kończy pracę. W przeciwnej sytuacji uruchamia ona procedurę oceny stwierdzeń dla każdego stwierdzenia wchodzącego w skład przesłanki. Po ustaleniu miar pewności i wątpliwości dla koniunkcji tych stwierdzeń obliczany jest stopień pewności przesłanki (CF). O ile jego wartość bezwzględna

przekracza pewną, progową wartość, następuje przekazanie sterowania do procedury wyciągającej wnioski z reguły. W zależności od znaku CF wnioskiem jest konkluzja prosta ( $CF > 0$ ) lub konkluzja przeciwna ( $CF < 0$ ). Gdy konkluzji odpowiedniego rodzaju (prostej lub przeciwniej) nie ma w danej regule, procedury wyciągającej wnioski się nie wywołuje.

Procedura wyciągająca wnioski kolejno przegląda wszystkie elementy konkluzji. Gdy element jest stwierdzeniem, jego stopień pewności jest aktualizowany zgodnie z równaniami (2.2). Dla stwierdzeń, które nie były jeszcze oceniane przyjmuje się początkowo  $MB = MD = 0$ . Gdy rozważany element wniosku jest komunikatem, zostaje on wyświetlony. Gdy zawiera on działania arytmetyczne, następuje ich wykonanie.

Jak widać procedury oceny stwierdzenia i oceny przesłanki wzajemnie się wykorzystują w sposób rekursywny.

Procedura prezentacji wyników służy do wyświetlenia końcowych wyników wnioskowania. Stanowiący rozwiązanie zestaw stwierdzeń, uszeregowany wg malejących wartości stopnia pewności (CF), jest wyświetlany na ekranie. Gdy nie wszystkie stwierdzenia mieszczą się w polu ekranu, możliwe jest przeglądanie zestawu stwierdzeń przy pomocy klawiszy  $\uparrow$  i  $\downarrow$ . W miarę możliwości, oprócz stopnia pewności (CF), podawane są jego składowe (MB i MD), co daje lepszą informację o "głosach za i przeciw" dla danego stwierdzenia.

Procedury objaśniające umożliwiają poznanie powodów, dla których uzyskano określone stwierdzenie. Korzystając z procedury prezentacji można ustawić wskaźnik na wybranym stwierdzeniu. Gdy stopień pewności wyświetlany jest w rozbiciu na składniki "za i przeciw", można przy pomocy klawiszy  $\leftarrow$  i  $\rightarrow$  ograniczyć zakres objaśnień. Wciśnięcie klawisza Enter powoduje uaktywnienie procedur objaśniania. Odnajdują one pierwszą regułę, która przyczyniła się do uzyskania analizowanego wniosku, wyświetlają jej numer oraz objaśniają-

cy komentarz (odczytany z pola 'Komentarz do wniosku' lub 'Komentarz do wniosku przeciwnego' relacji REGULY). Wciśnięcie klawisza PgDn powoduje wyświetlenie takiej informacji dla następnej reguły odpowiedzialnej za uzyskany wniosek (o ile taka istnieje). Posługując się klawiszami PgUp i PgDn można przeglądać wszystkie istotne reguły. Gdy potrzebne są bardziej szczegółowe objaśnienia, klawisz F2 umożliwia wyświetlanie stwierdzeń składających się na przesłankę rozważanej reguły. Odbywa się to przy pomocy zwykłych procedur prezentacji, tak więc możliwe jest następnie przejście do objaśniania tych stwierdzeń itd. Zatem procedury objaśniające i prezentujące wyniki również odwołują się do siebie wzajemnie, w rekursywny sposób.

#### 4.3.5. Komunikacja z użytkownikiem.

Komunikację z użytkownikiem realizują procedury obsługi ekranu, wyświetlania komunikatów, wyboru z "menu", edytory różnych typów danych itp.

Procedury obsługi ekranu stanowią jedyną grupę procedur, które są silnie zależne od sprzętu; wynika to stąd, że standard PASCALA nie przewiduje takich operacji jak definiowanie okienek, określanie kolorów, pozycjonowanie kursora itp. Wyświetlanie i redagowanie tekstu oraz wybór z "menu" obsługiwane są przez tę samą, dużą procedurę. Jej parametrami są deskryptor okienka oraz tryb pracy. Deskryptor okienka powinien oprócz współrzędnych ekranowych zawierać tekst, który ma być wyświetlony (i ewentualnie przetworzony). Tryb pracy określa operację jaką należy wykonać: wyświetlenie (DISP), redakcję (EDIT), pojedynczy wybór (MENU) i wybór wielokrotny (MULTMENU). Ta podstawowa procedura jest intensywnie używana przez wszystkie inne edytory, np. przez edytor danych w SZBD, czy edytor reguł systemu doradczego.

Procedury wyświetlania komunikatów wyświetlają na ekranie odpowiednie komunikaty lub realizują wybory z różnych "menu". Jednym z parametrów tych procedur jest numer komunikatu lub listy wyboru. Numer ten jest kluczem systemowej relacji zawierającej deskryptor okienka oraz potrzebny tekst. Za pośrednictwem procedur SZBD dane te są pobierane z bazy danych i przy użyciu procedur obsługi ekranu przedstawiane użytkownikowi. Przechowywanie komunikatów w bazie danych przynosi oszczędność pamięci operacyjnej zajmowanej przez program.

## 5. KONSTRUKCJA SYSTEMU

Po zrealizowaniu programu przystąpiono do tworzenia systemu informacji i systemu doradczego dotyczących własności i zasad stosowania leków hipotensyjnych, przeciwdusznicych i przeciwarystmicznych. Systemowi nadano nazwę KARDIO-LEK.

### 5.1. Konstrukcja bazy danych.

Posługując się zrealizowanym programem, zgodnie z założeniami podanymi w punkcie 3.5, zdefiniowano logiczną i fizyczną strukturę bazy danych. Następnie wprowadzono przygotowane dane [23], sprawdzono ich poprawność i usunięto dostrzeżone błędy. Baza danych jest stale aktualizowana, w miarę napływu nowych informacji. Wydruk bazy danych zamieszczony jest w pracy M. Wójcik-Jawień [23], na ryc. 6.2 znajduje się przykładowy opis jednego leku. Bieżący rozdział przedstawia wydruki ekranu komputera ilustrujące poszczególne fazy konstrukcji bazy danych.

#### 5.1.1. Określenie schematu i organizacji fizycznej.

Po przejściu do pracy w trybie administratora systemu i wybraniu operacji projektowania bazy danych, przedstawiono schemat bazy danych podając nazwy relacji, składniki klucza i listę pozostałych atrybutów (ryc. 5.1). Następnie określono organizację dostępu względem poszczególnych składników klucza, podano typy atrybutów, nazwy plików dyskowych i inne, wymagane przez program dane dotyczące fizycznej organizacji bazy danych (ryc. 5.2 i 5.3). Po zakończeniu projektowania relacji bazy danych rozplanowano formularz, który jest wykorzystywany przy wprowadzaniu i prezentacji danych. Kolejną czynnością było zainicjalizowanie relacji, po czym przystąpiono do wprowadzania danych.



Relacja: PODSTAWOWE\_WLASNOSCI

- KLUCZ GLOWNY -----
- > NAZWA\_MIEDZYNARODOWA
  - >
- POZOSTALE ATRYBUTY -----
- > NAZWA\_MIEDZYNAR\_PL
  - > NAZWA\_CHEMICZNA
  - > WZOR\_SUMARYCZNY
  - > MASA\_CZASTEczKOWA
  - > ROZPUSZCZALNOSC
  - > PH
  - > PKA
  - > TEMP\_TOPNIENIA
  - > WARUNKI\_PRZECHOWYWANIA
  - > GRUPA\_FARMAKOLOGICZNA
  - > MECHANIZM\_DZIALANIA
  - > DZIALANIE\_NA\_INNE\_UKLADY
  - > WSKAZANIA
  - > PRZECIWSKAZANIA\_WZGLEDNE
  - > PRZECIWSKAZANIA\_BEZWZGLEDNE

Rvc. 5.1. Projektowanie schematu logicznego bazy danych o leku.

Relacja : PODSTAWOWE\_WLASNOSCI

Obszar identyfikowany kluczem: NAZWA\_MIEDZYNARODOWA

Atrybut	Typ atrybutu
Atrybut : NAZWA_MIEDZYNAR_PL	Krotki tekst
Atrybut : NAZWA_CHEMICZNA	Dowolny tekst
>> Dlugosc wiersza tekstu: 60	Liczba calkowita
Atrybut : WZOR_SUMARYCZNY	Atrybut z ograniczonymi wartościami
Atrybut : MASA_CZASTEczKOWA	Typ wyliczeniowy
Atrybut : ROZPUSZCZALNOSC	Klucz innej relacji
>> Dlugosc wiersza tekstu: 60	Dane binarne
Atrybut : PH	Odnosniki
Atrybut : PKA	Ciagi wartosci
Atrybut : TEMP_TOPNIENIA	Zbiory

Rvc. 5.2. Projektowanie bazy danych o leku - okreslanie typu atrybutow.

PROJEKTOWANIE BAZY DANYCH Fizyczna organizacja relacji  
===== Relacja : DZIALANIE\_UBOCZNE =====

```
Obszar identyfikowany kluczem:UKLAD
Atrybut : UKLAD
>> Typ wyliczeniowy: _UKLADY
>>> Wymien składniki typu w kolejności
    * POKARMOWY
    * NERWOWY
    * KRAZENIA
    * KREW
    * MOCZ
    * SKORA
    * DOKREWNY
    * WYDALNICZY
    * ODDECHOWY
    * INNE
    *
Atrybut : OPIS_DZIALANIA
>> Długość wiersza tekstu: 68
****
> Przewidywana ilość różnych wartości
> klucza na tym poziomie dostępu:5
> Plik dyskowy z obszarem nadmiarowym: LEK.DAT
```

Ryc. 5.3. Projektowanie bazy danych o leku - elementu organizacji fizycznej.

### 5.1.2. Wprowadzanie danych.

Po wybraniu (z odpowiedniego "menu") relacji, która ma być aktualizowana, można, korzystając z edytora danych wpisywać do bazy danych odpowiednie informacje (Ryc. 5.4). W ten sposób zgromadzono cały zasób informacji o rozważanych lekach.

ADMINISTRATOR

Redagowanie danych

NAZWA\_MIEDZYNARODOWA  
LANATOSIDE C

NAZWA\_MIEDZYNAR.(PL)  
Lanatozyd C

NAZWA\_CHEMICZNA

3beta.12beta.14beta-trihydroksy-5beta-kard-20(22)-enolido-  
3-monoacetylotridigitoksozylukozyd

WZOR\_SUMARYCZNY  
C49 H76 O28

MASA\_CZASTECZKOWA  
985.1

ROZPUSZCZALNOSC

Rozpuszczalny (1cz.) w 45cz. etanolu, 2000cz. chloroformu,  
20cz. metanolu, praktycznie nierozpuszczalny w eterze  
i w wodzie.

! - wybór atrybutu Esc - koniec Ins - poprawki Ime - nowa wartość

Rvc. 5.4. Wprowadzanie informacji o leku do bazy danych.

## 5.2. Konstrukcja systemu doradczego.

Wykorzystując przygotowany moduł do budowy systemu ekspertowego utworzono bazę wiedzy o lekach krążeniowych. Wcześniej, przy użyciu SZBD, uzupełniono bazę danych o relację WYWIAD, której projekt przedstawiono w punkcie 3.6. Proces tworzenia bazy wiedzy przebiegał iteracyjnie, tj. po wprowadzeniu pewnej ilości reguł dotyczących określonego problemu, sprawdzano prawidłowość funkcjonowania systemu i korygowano błędy, po czym wprowadzano dalsze reguły itd. Kompletną bazę wiedzy testowano w obecności lekarzy, dokonując odpowiednich korekt, stosownie do ich uwag. Również napływające z piśmiennictwa nowe informacje związane z wiedzą systemu są stale wykorzystywane do aktualizacji i strojenia systemu.

Poniżej przedstawiono wydruki ilustrujące proces konstrukcji bazy wiedzy systemu doradczego.

### 5.2.1. Redagowanie bazy wiedzy.

Redagowanie bazy wiedzy polega na wpisywaniu nowych reguł bądź modyfikacji istniejących. W tym celu użytkownik po wybraniu trybu pracy 'Administrator' i przejściu do redagowania bazy wiedzy obowiązany jest podać numer reguły. Gdy numer ten wskazuje na istniejącą regułę jest ona wyświetlana na ekranie i użytkownik może dokonywać w niej potrzebnych zmian. Gdy reguły o wybranym numerze nie ma w bazie wiedzy, edytor umożliwia jej utworzenie. Redagowanie reguł polega na wpisaniu przesłanki, konkluzji (ew. także wniosku przeciwnego) i podaniu stopnia pewności stwierdzeń występujących we wnioskach (ryc. 5.5). Czynnością uzupełniającą jest podanie używanych w procesie objaśniania komentarzy do wniosków (ryc. 5.6).



## 6. WYNIKI

### 6.1. Przykłady ilustrujące działanie systemu.

W niniejszym punkcie zostaną przedstawione reprodukcje ekranu komputera ilustrujące funkcjonowanie systemu.

#### 6.1.1. Wyszukiwanie informacji.

Poniższe ryciny przedstawiają kilka przykładów wykorzystania systemu do wyszukania odpowiednich informacji.

##### 6.1.1.1. Uzyskanie informacji o konkretnym leku.

Ryc. 6.1 przedstawia zapytanie o pełną informację na temat amiodaronu. Zażądano wydrukowania odpowiedzi na drukarce. Uzyskany wydruk przedstawia ryc. 6.2.

Pytanie o preparat o nazwie FURIX i uzyskaną na ekranie odpowiedź przedstawiają ryc. 6.3 i 6.4.

```
..... WYSZUKIWANIE ..... Redagowanie pytania .....
NAZWA_MIEDZYNARODOWA
NAZWA_MIEDZYNAR.(PL)      Rowne Amiodaron
NAZWA_CHEMICZNA
WZOR_SUMARYCZNY
MASA_CZASTEczKOWA
ROZPUSZCZALNOSC
PH
PKA
TEMP.TOPNIENIA
WARUNKI_PRZECHOWYWANIA
GRUPA_FARMAKOLOGICZNA
MECHANIZM_DZIALANIA
DZIALANIE_NA_INNE_UKLADY
WSKAZANIA
PRZECIWSK.WZGLEdNE
PRZECIWSK.BEZWZGLEdNE
.....

f1 - wybór atrybutu   Esc - koniec   inne - redakcja wzorca
```

Ryc. 6.1. Pytanie o amiodaron

NAZWA\_MIEDZYNARODOWA  
AMIODARONE HYDROCHLORIDE

NAZWA\_MIEDZYNAR.(PL)  
Amiodaron

NAZWA\_CHEMICZNA

chlorowodorek (2-butyl-3-benzofuranylo)-(4-[2-(dietyloamino-  
-etoksy]-3,5-dijodofenylo) metanonu

WZOR\_SUMARYCZNY

C<sub>25</sub> H<sub>29</sub> J<sub>2</sub> N O<sub>3</sub>. HCl

MASA\_CZASTECZKOWA

681.8

ROZPUSZCZALNOSC

Dobrze rozpuszcza sie w chloroformie (44.5g/100ml), slabo  
w etanolu (1.3g/100ml), bardzo slabo w wodzie (0.07g/100ml).

PH

3.0

PKA

6.5

TEMP.TOPNIENIA

159.0

WARUNKI\_PRZECHOWYWANIA

GRUPA\_FARMAKOLOGICZNA

Antiarrhythmicum, Vasodilatans.

MECHANIZM\_DZIALANIA

Grupa III wg Williama ( wydłużenie czasu potencjalu czyn-  
nościowego, wydłużenie skutecznego okresu refrakcji).  
Zaznaczone działanie alfa- i beta-adrenolityczne.  
Bezpośredni wpływ na mięśniówkę naczyń.

DZIALANIE\_NA\_INNE\_UKLADY

Zaznaczone działanie hipotensyjne.

WSKAZANIA

napadowy częstoskurcz nadkomorowy, napadowe migotanie przed-  
sionków, utrwalone migotanie przedsionków, ekstrasystolia  
komorowa, napadowy częstoskurcz komorowy, zespół WPW, zabu-  
rzenia rytmu w zawał serca, choroba wieńcowa,

PRZECIWSK.WZGLEDNE

dychawica oskrzelowa

PRZECIWSK.BEZWZGLEDNE

bradykardia, blok SA, AV lub blok odnog pęczka Hisa, ciąża,  
W zapaści sercowo-naczyniowej i hipotonii nie stosować i.v.

DZIALANIE UROCZNE

Układ	Opis
POKARMOWY	brak łaknienia, nudności, zaparcia,
NERWOWY	ból głowy, drżenia mięśniowe, bezsenność, neuropatia obwodowa,

KRAZENIA	hipotonia, bradykardia,
KREW	hipoglikemia, hiperinsulinemia, wzrost aktywnosci AlAT i AspAT,
SKORA	szaroniebieska pigmentacja skory, nadwrazliwosc na slonce,
DOKREWNY	nadczynnosc lub niedoczynnosc tarczycy,
ODDECHOWY	srodmiaszczowe zapalenie pluc,
INNE	impotencja

#### INTERAKCJE

Lek 1	Lek 2	Opis interakcji
AMIODARONE HYDROCHLORIDE	ANAESTHETICUM	Hipotonia i bradykardia oporna na atropine.
AMIODARONE HYDROCHLORIDE	ANTIARRHYTHMICUM	Kojarzenie z lekami nalezacyimi do grupy I wg Williama moze wywolac arytmie komorowa typu torsades de pointes.
AMIODARONE HYDROCHLORIDE	DIGOXIN	Wzrost poziomu digoksyny o 70-300% .
AMIODARONE HYDROCHLORIDE	VERAPAMIL HYDROCHLORIDE	Zaburzenia elektrofizjologiczne.
AMIODARONE HYDROCHLORIDE	DILTIAZEM HYDROCHLORIDE	Zaburzenia elektrofizjologiczne.
AMIODARONE HYDROCHLORIDE	PROCAINAMIDE HYDROCHLORID	Nasilenie dzialania inotropowo-ujemnego.
AMIODARONE HYDROCHLORIDE	QUINIDINE SULFATE	Wzrost poziomu prokainamidu o 55% .
AMIODARONE HYDROCHLORIDE	PHENAZONE	Wzrost poziomu chinidyny o 33% .
AMIODARONE HYDROCHLORIDE	WARFARIN SODIUM	Wzrost poziomu antypiryny.
AMIODARONE HYDROCHLORIDE	NIFEDIPINE	Dwukrotny wzrost poziomu amiodaronu, trzykrotny wzrost czasu protrombinowego.
AMIODARONE HYDROCHLORIDE	DISOPYRAMIDE	Korzystne polaczenie w dusznicy bolesnej.
AMIODARONE HYDROCHLORIDE	PROPAFENONE HYDROCHLORIDE	Unikac lacznego podawania, z uwagi na nasilenie dzialan niepozadanych.
AMIODARONE HYDROCHLORIDE	PROPAFENONE HYDROCHLORIDE	Nasilenie dzialania propafenonu.

#### FARMAKOKINETYKA

Parametr	Opis
DOSTEPNOSC_BIOLOGICZNA	22 - 86 %
OKRES_POLTRWANIA	3.2 - 80 h ; wzrasta do 100 dni po dlugotrwej terapii
KRT	6.3 - 143 h
STEZENIE_TERAPEUTYCZNE	0.4 - 12 ug/ml; nalezy utrzymywac poziom 1 - 2.5 ug/ml
WIAZANIE_Z_BIALKAMI	96.3 +/- 0.6 %
OBJETOSC_DYSTRYBUCJI	43 - 1151 l
KLIRENS_OGOLNOUSTROJOWY	100 - 800 ml/min
KLIRENS_NERKOWY	1 - 8 ml/min
WYDALANIE_Z_MOCZEM	ponizej 1 % w formie niezmienionej
AKTYWNE_METABOLITY	brak
PRZECH_PRZEZ_LOZYSKO	przechodzi; stezenie w krwi plodu stanowi 25 % stezenia w krwi matki; lek moze byc teratogeny ze wzgledu na duza zawartosc jodu
PRZECH_DO_POKARMU_KOB	przechodzi; stezenie w pokarmie jest 2 - 9 razy wyzsze od stezenia w krwi
DIALIZA	nie dializuje

#### DAWKOWANIE

Ze wzgledu na duze roznicze osobnicze w farmakokinytyce leku dawka powinna byc ustalana indywidualnie (po oznaczeniu jego stezenia w krwi). Zwykle stosuje sie:

i.v. : 5 mg/kg przez 3 minuty lub we wlewie po rozcienczeniu 300 mg w 250 ml 5 % roztworu glukozy w ciagu 20 min - 2 h, powtarzajac wlew 2 - 3 razy na dobe.

p.o. : 600 - 800 mg/dzien przez 8 - 10 dni podczas posilkow  
W leczeniu podtrzymujacym 200 - 400 mg/dzien przez



20 dni w miesiącu.

PREPARATY\_DOSTEPNE\_W\_KRAJU

CORDARONE - tabl. 0.2 g, cena: 5400 zł/60 szt.  
inj. 0.15 g/3ml, cena: 300 zł/ 1 amp.

NAZWY

Nazwa zastrzeżona	Firma	Kraj
CORDAREX	LABAZ	D
CORDARONE	LABAZ	F
CORDARONE	LABAZ	GB
CORDARONE	KRKA	YU
CORBIONAX	ROLAND-MARIE	F

UWAGI

- Bardzo duże różnice osobnicze w farmakokinetyce leku.
- W okresie leczenia należy chronić się przed działaniem promieni słonecznych.

PISMIENICTWO

- Adamska-Dyniewska H.: "Amiodaron - mechanizmy działania i zastosowanie kliniczne"; Pol.Tyg.Lek. 39,1707,1984.
- Gillis A.M., Kates R.E.: "Clinical Pharmacokinetics of the Newer, Antiarrhythmic Agents"; Clin. Pharmacokin. 9,375,1984.
- Gould L.A.: "Drug Treatment of Cardiac Arrhythmias", Futura Publishing Company, Mount Kisco, New York 1983.
- Harrison D.C.: "Cardiac Arrhythmias a Decade of Progress" G.K. Hill Medical Publishers, Boston, Massachusetts 1981.
- Keefe D.L.D. et al.: "New Antiarrhythmic Drugs: Their Place in Therapy"; Drugs, 22,363,1981.
- Latini R. et al.: "Clinical Pharmacokinetics of Amiodarone"; Clin.Pharmacokin. 9,136,1984.
- Martindale the Extra Pharmacopeia, 28Ed., Pharm. Press, London, 1982.
- Morganroth J.: "Comparative Evaluation of Antiarrhythmic Agents", Drugs, 29,suppl.4,14,1985.
- Podlewski J.K., Chwałibogowska-Podlowska A.: "Leki współczesnej terapii", PZWL, W-wa 1985.
- Plomp T.A. et al.: "Pharmacokinetics and Body Distribution of Amiodarone in Man"; Arzneim.Forsch. 34,513,1984.
- Rydlewska-Sadowska W., Sadowski Z.: "Zaburzenia rytmu serca", PZWL, W-wa 1985.
- Schepper M., Olsson B.: "Cardiac Arrhythmias. Diagnosis, Prognosis, Therapy", Springer-Verlag, Berlin Heidelberg, New York 1983.
- Schwartz J.B. et al.: "Adverse Effects of Antiarrhythmic Drugs", Drugs 21,23,1981.
- Staeubli M. et al.: "Serum Concentrations of Amiodarone During Long Term Therapy. Relation to Dose, Efficacy and Toxicity"; Eur.J.Clin.Pharmacol. 24,485,1983.
- Tucker G.T. et al.: "Bioavailability of Amiodarone"; Eur.J.Clin.Pharmacol. 26,533,1984.
- Wetterell G., Andersson K.E.: "Cardiovascular Drugs I.: Antidysrhythmic Drugs", Ther.Drug.Monitor. 8,59,1986.

WYSZUKIWANIE	Redagowanie pytania
Nazwa międzynarodowa	
Nazwa zastrzeżona	Rowne      FURIX
Firma	
Kraj	



Ryc. 6.3. Pytanie o preparat o nazwie zastrzeżonej FURIX.

WYSZUKIWANIE	Prezentacja
Nazwa międzynarodowa FUROSEMIDE	
Nazwa zastrzeżona FURIX	Firma BENZON
	Kraj S

PgUp/PgDn - przeglądanie wyników      Esc - koniec

Ryc. 6.4. Przedstawienie wyników wyszukiwania na ekranie.

### 6.1.1.2. Wyszukiwanie z wyborem interesujących atrybutów.

Pytanie: które leki są wskazane w zespole WPW? Podać jakie mają przeciwwskazania.

Po sformułowaniu pytania (ryc. 6.5) użytkownik wskazuje, które atrybuty go interesują (ryc. 6.6). Dla kontroli polecono również drukować atrybut WSKAZANIA. Fragment uzyskanej odpowiedzi zawiera wydruk na ryc. 6.7.

WYSZUKIWANIE	Redagowanie pytania
NAZWA_MIEDZYNARODOWA	
NAZWA_MIEDZYNAR.(PL)	
NAZWA_CHEMICZNA	
WZOR_SUMARYCZNY	
MASA_CZASTECzkOWA	
ROZPUSZCZALNOSC	
PH	
PKA	
TEMP.TOPNIENIA	
WARUNKI_PRZECHOWYWANIA	
GRUPA_FARMAKOLOGICZNA	
MECHANIZM_DZIALANIA	
DZIALANIE_NA_INNE_UKLADY	
WSKAZANIA	Zawiera WPW
PRZECIWWSK.WZGLEDNE	
PRZECIWWSK.BEZWZGLEDNE	

F1 - wybór atrybutu Esc - koniec inne - redakcja wzorca

Ryc. 6.5. Pytanie o leki wskazane w zespole WPW.

WYSZUKIWANIE Redagowanie pytania

NAZWA_MIEDZYNARODOWA	
NAZWA_MIEDZYNAR.(PL)	
NAZWA_CHEMICZNA	
WZOR_SUMARYCZNY	
MASA_CZASTEczKOWA	
ROZPUSzcZALNOSC	
PH	PH
PKA	PKA
TEMP.TOPNIENIA	TEMP.TOPNIENIA
WARUNKI_PRZECHOWYWANIA	WARUNKI_PRZECHOWYWANIA
GRUPA_FARMAKOLOGICZNA	GRUPA_FARMAKOLOGICZNA
MECHANIZM_DZIALANIA	MECHANIZM_DZIALANIA
DZIALANIE_NA_INNE_UKLADY	DZIALANIE_NA_INNE_UKLADY
WSKAZANIA	WSKAZANIA J
PRZECIwWSK.WZGLEdNE	PRZECIwWSK.WZGLEdNE J
PRZECIwWSK.BEZwZGLEdNE	PRZECIwWSK.BEZwZGLEdNE J
NAZWA_MIEDZYNARODOWA	
NAZWA_MIEDZYNAR.(PL)	J
NAZWA_CHEMICZNA	
WZOR_SUMARYCZNY	
MASA_CZASTEczKOWA	

f1 - lub odstep    ruch wskaznika    < - wybor atrybutu    Esc - gotowe

Ryc. 6.6. Określanie wykazu potrzebnych atrybutów.

NAZWA\_MIEDZYNAR.(PL)

Digoksyna

WSKAZANIA

przewlekła niewydolność lewo- i prawokomorowa, ostra niewydolność lewokomorowa, po operacjach na sercu szczególnie wad zastawkowych, napadowe migotanie i trzepotanie przedsionków z częstoskurczem, napadowe częstoskurcze nadkomorowe i węzłowe, napadowe częstoskurcze nadkomorowe w zespole WPW, dusznica bolesna (u chorych ze zmniejszoną objętością wyrzutowa serca i częstoskurczem).

PRZECIWNISK.WZGLEDNE

niewydolność oddechowa (znacznego stopnia), niewydolność nerek (konieczność monitorowania poziomu w krwi), nadciśnienie tętnicze (przed jego obniżeniem), zespół chorego węzła zatokowego, ekstrasystole komorowe, bradykardia i zaburzenia przewodzenia w czasie stosowania leków przeciwarrytmicznych, przed planowaną kardiowersją elektryczną,

PRZECIWNISK.BEZWZGLEDNE

blok AV II i III st. szczególnie z napadami MAS, zaciśkowa przerostowa kardiomiopatia, objawy zatrucia glikozydami narciowymi, częstoskurcz komorowy i migotanie komor, zespół WPW z napadowymi częstoskurczami komorowymi,

NAZWA\_MIEDZYNAR.(PL)

Propafenon

WSKAZANIA

ekstrasystolia komorowa, napadowy częstoskurcz komorowy, napadowy częstoskurcz nadkomorowy, trzepotanie przedsionków, napadowe migotanie przedsionków, zespół WPW, zaburzenia rytmu w zawale serca.

PRZECIWNISK.WZGLEDNE

niewydolność wątroby lub nerek

PRZECIWNISK.BEZWZGLEDNE

nieleczona niewydolność mięśnia sercowego, wstrząs kardiogeny, silna bradykardia, zaburzenia przewodnictwa zatokowego, przedsionkowo-komorowego i wewnątrzkomorowego, ciężkie schorzenia płuc z niedrożnością, hipotonia, zaburzenia w gospodarce elektrolitowej, pierwszy trymestr ciąży i okres karmienia.

NAZWA\_MIEDZYNAR.(PL)

Dizopiramid

WSKAZANIA

ekstrasystolia komorowa, napadowy częstoskurcz komorowy, zespół WPW, napadowy częstoskurcz nadkomorowy, trzepotanie przedsionków, napadowe migotanie przedsionków, ekstrasystolia węzłowa, zaburzenia rytmu w zawale serca,

PRZECIWNISK.WZGLEDNE

PRZECIWNISK.BEZWZGLEDNE

blok AV II i III st., choroba węzła SA z utratami przytomności, niewydolność lewokomorowa, skazy krwotoczne, jaskra, alergia na lek, hiperkaliemia, przerost prostaty, ciąża, miasthenia gravis,

NAZWA\_MIEDZYNAR.(PL)  
Apyryndyna

WSKAZANIA

zespół WPW, napadomy częstoskurcz komorowy, ekstrasystolia komorowa, napadome migotanie przedsionków, napadomy częstoskurcz nadkomorowy,

PRZECIWNISK.WZGLEDNE

PRZECIWNISK.BEZWZGLEDNE

ciężka niewydolność wątroby, nerek lub mięśnia sercowego, całkowity blok AV, hipotonia, bradykardia choroba Parkinsona padaczka, ciąża, okres aktywności płciowej kobiet.

NAZWA\_MIEDZYNAR.(PL)  
Amiodaron

WSKAZANIA

napadomy częstoskurcz nadkomorowy, napadome migotanie przedsionków, utrwalone migotanie przedsionków, ekstrasystolia komorowa, napadomy częstoskurcz komorowy, zespół WPW, zaburzenia rytmu w zawał serca, choroba wieńcowa,

PRZECIWNISK.WZGLEDNE

dychawica oskrzelowa

PRZECIWNISK.BEZWZGLEDNE

bradykardia, blok SA, AV lub blok odnog peczka Hisa, ciąża, w zapadzi sercowo-naczyniowej i hipotonii nie stosować i.v.

NAZWA\_MIEDZYNAR.(PL)  
Ajmalina

WSKAZANIA

napadome migotanie przedsionków w zespole WPW, trzepotanie przedsionków w zespole WPW, ekstrasystolia komorowa, napadomy częstoskurcz komorowy, rozpoznawanie napadomego bloku AV, zespołu WPW i LGL.

PRZECIWNISK.WZGLEDNE

zawał mięśnia sercowego, niewydolność krążenia, podeszły wiek,

PRZECIWNISK.BEZWZGLEDNE

blok AV II i III st., napadomy blok AV, choroba węzła AV z utratami przytomności, środkomrowe zaburzenia przewodzenia, kardiomegalia, zaawansowana niewydolność krążenia, wstrząs, hipotonia, torsades de pointe, zespół wydłużonego QT, ciąża,

### 6.1.2. Wybór leku przez system doradczy.

Ryciny zamieszczone w tym punkcie ilustrują cały cykl funkcjonowania systemu doradczego - od wstępnego wywiadu lekarskiego, poprzez uzupełniającą konwersację, aż po prezentację i objaśnianie wyników. Ekspertyza jest wykonywana dla jednej z pacjentek Kliniki Geriatrii AM w Krakowie.

#### 6.1.2.1. Wywiad lekarski.

Pacjentka B.M., 1.78, cierpiąca na zaburzenia rytmu serca (migotanie przedsionków), nadciśnienie tętnicze i dusznicę bolesną, z objawami przewlekłej niewydolności krążenia i stwierdzoną niewydolnością wątroby. Pierwszą czynnością przy pracy z systemem doradczym jest wpisanie podstawowych danych o pacjentce. Ryc. 6.8 zawiera wydruk danych z tego wstępnego wywiadu.

#### 6.1.2.2. Analiza przypadku.

Pierwsze pytanie dotyczyło wyboru leku przeciwaritmicznego. Po sformułowaniu tego pytania przez użytkownika system przystępuje do analizy problemu. Pierwsze pytanie zadane użytkownikowi dotyczy rodzaju arytmii (ryc. 6.9). Użytkownik odpowiada, zaznaczając stwierdzoną arytmie (utrwalone migotanie przedsionków). W drugim pytaniu komputer w podobny sposób żąda określenia stopnia wydolności krążenia. Lekarz wybiera odpowiedź *NYHA 3*. Następne pytanie systemu dotyczy stopnia upośledzenia funkcji wątroby (ryc. 6.10). Odpowiedź polega na określeniu, przy pomocy specjalnego wskaźnika, stopnia tej niewydolności. Lekarz ocenia stopień niewydolności wątroby jako raczej niewielki.

Symbol identyfikacyjny  
B.M.

Plec  
KOBIEȦA

Wiek [lata]  
78

Wzrost [cm]  
164

Waga [kg]  
55.0

Tetno [uderz/min]  
78

Cisnienie skurczowe [mm Hg]  
170

Cisnienie rozkurczowe [mm Hg]  
85

Schorzenie, ktore ma byc leczone  
ZABURZENIA\_RYTMU\_SERCA  
NADCISNIENIE\_TETNICZE\_SANDISTNE  
DUSZYNICA\_BOLESNA

Schorzenia towarzyszące  
PRZEWLEKLA\_NIEWYDOLNOSC\_KRAZENIA  
NIEWYDOLNOSC\_WATROBY

Przebyte schorzenia  
NIEWYDOLNOSC\_WATROBY

Glukoza [mmol/l]  
5.4

Cholesterol [mmol/l]  
5.7

Kreatynina [umol/l]  
70.6

Potas [mmol/l]  
4.5

Kwas moczowy [mmol/l]  
0.2



WNIOSKOWANIE		Analiza
Sformulowanie		problemu
C O	JEST	WSK.LEKIEM P-ARYTMICZNYM
Pytania systemu		
C Z Y		
RODZAJ ARYTMII		
Z reguły ZABURZEN	NIEWYDOLNOSC_WEZLA_SA	
	TRZEPOTANIE_PRZEDSIONKOW	
	NAPADOWE_MIGOTANIE_PRZEDSIONKOW	
	UTRWALONE_MIGOTANIE_PRZEDSIONKOW	
	WIELOKSIET_CZESTOSKURCZ_PRZEDSIONKOW	
	EXSTRASYSTOLIA_WEZLOWA	
	NIE NAPADOWY_CZESTOSKURCZ_WEZLOWY	
		STWIERDZONYM SCHORZENIEM

Rvc. 6.9. Użytkownik odpowiada na zadane przez system doradczy pytanie o rodzaj arytmii.

WNIOSKOWANIE		Analiza
Sformulowanie		problemu
C O	JEST	WSK.LEKIEM P-ARYTMICZNYM
Pytania systemu		
C Z Y		
UPOSLĘDZENIE FUNKCJI WĄTROBY	JEST	ZNACZNE
		N
	stopień pewności	
	nie	na pewno
Reguła 154 przemawia za tym (BRETYLIUM)	wiem : [ ] : NIE	
Reguła 154 przemawia za tym (WERAPAMIL)	chyba "raczej"	AZANYM LEKIEM
Reguła 154 przemawia za tym (GLIKOZYDY NAPARSTNICZY)	NIE NIE	AZANYM LEKIEM
	SA	WSKAZANA GRUPA LEKOW
Z reguły 528 wynika (-0.98), za: UPOSLĘDZENIE KRAZENIA OBWODOWEGO	NIE JEST	STWIERDZONYM SCHORZENIEM
Z reguły 538 wynika (-0.98), za: STANY BRONCHOSPASTYCZNE	NIE JEST	STWIERDZONYM SCHORZENIEM

T - Tak N - Nie ? - określenie stopnia pewności

Rvc. 6.10. Użytkownik ocenia stopień niewydolności funkcji wątroby przy pomocy ruchomego wskaźnika.

Dalszy dialog lekarza (l) i komputera (k) można streścić następująco:

k: Czy stwierdzono blok AV ?  
l: *nie*  
k: Określ postać dusznicy.  
l: *wysiłkowa*  
k: Określ postać nadciśnienia.  
l: *łagodne*  
k: Określ stadium ndciśnienia.  
l: *stadium II.*  
k: Czy pacjent był już leczony na nadciśnienie ?  
l: *tak*  
k: Czy znany jest trzykrotny pomiar ciśnienia ?  
l: *tak*  
k: Podaj wartości ciśnienia skurczowego w mmHg.  
l: *170, 150, 140,*  
k: Podaj wartości ciśnienia rozkurczowego w mmHg.  
l: *90, 85, 90.*

#### 6.1.2.3. Prezentacja i objaśnianie rozwiązań.

Teraz komputer prezentuje uzyskane przezeń rozwiązanie (ryc. 6.11).

W podobny sposób użytkownik pyta o wskazany lek hipotensyjny. Dialog jest teraz krótki, gdyż większość potrzebnych informacji system uzyskał już poprzednio:

k: Określ aktywność reniny.  
l: *nieznana*  
k: Czy stwierdzono lewokomorową niewydolność serca ?  
l: *tak*  
k: Czy stwierdzono obrzęki ?  
l: *tak*  
k: Czy stwierdzono skłonność do zakrzepów ?  
l: *nie.*

Leki hipotensyjne wybrane przez system przedstawia ryc. 6.12.

WNIOSKOWANIE	Sformułowanie	Prezentacja problemu
C 0	JEST	WSK. LEKIEM P-ARYTMICZNYM
Znalezione rozwiązanie:		
DIGOKSYNA	JEST	WSK. LEKIEM P-ARYTMICZNYM St. pewności: 68.88 %
LAMATOZYD C	JEST	WSK. LEKIEM P-ARYTMICZNYM St. pewności: 60.08 %
DIGITOKSYNA	JEST	WSK. LEKIEM P-ARYTMICZNYM St. pewności: 65.88 %
AMIODARON	JEST	WSK. LEKIEM P-ARYTMICZNYM St. pewności: 43.69 %
ATENOLOL	JEST	WSK. LEKIEM P-ARYTMICZNYM St. pewności: 27.98 %
CHINIDYNA	NIE JEST	WSK. LEKIEM P-ARYTMICZNYM St. pewności: 21.85 %

f1 - wybór wniosku    ↔ - za/przeciw    <- - wyjaśnienie

Rvc. 6.11. Wykaz leków, badawcy odpowiedzieli systemowi doradczemu na pytanie o wskazany lek przeciwarrytmiczny.

WNIOSKOWANIE	Sformułowanie	Prezentacja problemu
C 0	JEST	WSK. LEKIEM HIPOTENSYJNYM
Znalezione rozwiązanie:		
NIFEDYPINA	JEST	WSK. LEKIEM HIPOTENSYJNYM St. pewności: 86.22 %
ENALAPRYL	JEST	WSK. LEKIEM HIPOTENSYJNYM St. pewności: 77.98 %
NITRENDYPINA	JEST	WSK. LEKIEM HIPOTENSYJNYM St. pewności: 66.98 %
KAPTOPRYL	JEST	WSK. LEKIEM HIPOTENSYJNYM St. pewności: 58.98 %
FUROSEMID	JEST	WSK. LEKIEM HIPOTENSYJNYM St. pewności: 58.35 %
BUMETANID	JEST	WSK. LEKIEM HIPOTENSYJNYM St. pewności: 58.35 %
MUZOLIMINA	JEST	WSK. LEKIEM HIPOTENSYJNYM St. pewności: 58.35 %
MEFIZYD	JEST	WSK. LEKIEM HIPOTENSYJNYM St. pewności: 58.35 %
PIRETANID	JEST	WSK. LEKIEM HIPOTENSYJNYM

f1 - wybór wniosku    ↔ - za/przeciw    <- - wyjaśnienie

Rvc. 6.12. Odpowiedź na pytanie o wskazane leki hipotensyjne.

Użytkownik, chcąc poznać przyczynę przedstawionej oceny atenololu, wskazuje ten lek i żąda wyjaśnienia. W odpowiedzi system wyświetla komentarze do tych reguł, które wypowiedzają się na temat wybranego leku i zostały skutecznie użyte w procesie wnioskowania (ryc. 6.13).

WNISKOWANIE	Objasnienie
Sformułowanie	problemu
C 0	JEST WSK. LEKIEM P-ARYTMICZNYM
Znalezione rozwiązanie:	
ATENOLOL	JEST WSKAZANYM LEKIEM
Za: 00.22 x	Przeciw: 21.02 x St. pewności: 22.08 x

Regula nr 185

Beta-blokery wykazują mniejszą skuteczność u pacjentów w podeszłym wieku. Stosowanie ich ograniczają liczne objawy uboczne. Bezpieczniej jest podawać beta-blokery kardiowybiórcze i/lub obdarzone wewnętrzną aktywnością sympatolityczną.

PgDn - następna, PgUp - poprzednia reguła, F2 - przesłanki, Esc - koniec

Ryc. 6.13. Objasnianie - prezentacja jednej z reguł, która wypowiedzała się przeciw użyciu atenololu.

## 6.2. Ocena programu.

Zgodnie z przyjętymi założeniami, THESIS daje użytkownikowi możliwość tworzenia i eksploatacji informacyjnej bazy danych oraz re-  
gułowego systemu doradczego. W programie zastosowano dobrze znane  
użytkownikom mikrokomputerów środki komunikacji komputer-człowiek,  
takie jak dokonywanie wyboru z "menu", nakładane okienka, wiersz  
podpowiedzi, wiersz stanu, wykorzystanie dźwięku, kolorów itp., co  
wraz z częściową kontrolą poprawności wprowadzanych danych i odpo-  
wiedzi oraz dążeniem do osiągnięcia całkowitej odporności na błędy  
popelniane przez użytkownika, nadaje mu cechy tzw. programu "przy-  
jaznego". Uzupełnienie programu o rozbudowany samouczek (ang.  
*help*) i usunięcie istniejących jeszcze niesprawności korzystnie  
wpłynęłoby na jego jakość.

Poniżej przeprowadzone jest porównanie zrealizowanych elementów  
programu z odpowiednimi założeniami projektu.

### 6.2.1. System Zarządzania Bazą Danych (SZBD).

Przedstawione w punkcie 3.3 założenia projektowe SZBD zostały  
na ogół zrealizowane. Bazy danych tworzone pod kontrolą programu  
THESIS mają strukturę relacyjną i mogą składać się z wielu relacji,  
co umożliwia uzyskanie potrzebnego stopnia normalizacji. Zestaw  
dostępnych typów danych jest bogatszy od spotykanego w popularnych  
firmowych SZBD (takich jak np. dBase III). Nie zrealizowano jednak  
typu "ciąg elementów" i specjalnego typu, pozwalającego na oszczęd-  
ne przechowywanie takich danych jak odnośniki do literatury. Spo-  
sób redagowania zapytań nie wymaga poznawania specjalnego języka  
i nie powinien nastręczać zasadniczych trudności. Edytor danych o-  
kazał się wygodnym narzędziem aktualizacji BD; szczególnie pożyte-  
czną cechą jest istnienie wartości domyślnych. Przewidziane proje-  
ktem organizacje fizyczne zostały zrealizowane, z tym jednak wyjąt-  
kiem, że organizację rozproszoną można stosować tylko na najwyższym

poziomie dostępu (tj. może ona dotyczyć tylko tego składnika klucza, który zajmuje pierwsze miejsce na liście dostępu (por. 4.2.2)). Odczuwa się również potrzebę wprowadzenia innych organizacji (np. indeksowo-sekwencyjnej) z powodu na znanych niedostatków organizacji rozproszonej (konieczność deklarowania maksymalnej ilości krotek i przypadkowy porządek kluczy w pliku).

Spośród wszystkich elementów programu najwięcej do życzenia pozostawia edytor struktury BD, zwłaszcza ze względu na trudność modyfikacji już istniejących relacji. W czasie projektowania organizacji fizycznej określa się tzw. "sposób kodowania atrybutu". Występuje przy tym pomieszczenie elementów logicznej i fizycznej struktury bazy danych, co zaburza postulowaną niezależność tych struktur. Pomimo tego, zmiany organizacji logicznej, czy fizycznej na ogół nie pociągały za sobą konieczności zmian w programie (i odwrotnie). Brak środków blokowania zapisu ogranicza możliwość sieciowego wykorzystania bazy danych jedynie do odczytu informacji.

Rozbudowy wymaga język zapytań. O ile łatwo jest wyrazić w nim proste pytania wyszukiwawcze, o tyle brak możliwości formułowania pytań zawierających alternatywę (spójnik "LUB") i inne ograniczenia utrudniają uzyskanie odpowiedzi na bardziej złożone pytania.

Brak możliwości stosowania liter charakterystycznych dla języka polskiego (ą, ę itd.) nie prowadził wprawdzie do nieporozumień, jest jednak istotną wadą.

#### 6.2.2. Moduły tworzenia i eksploatacji systemu doradczego.

THESIS umożliwia reprezentowanie wiedzy w postaci zbioru reguł, a przyjęta składnia zapisu stwierdzeń pozwala na wyrażanie ich w postaci zdań, które na ogół brzmią poprawnie po polsku, a przynajmniej powinny być zrozumiałe. Edytor bazy wiedzy i maszyna wnioskująca uwzględniają stopień pewności reguł i stwierdzeń; daje to także ograniczoną możliwość tworzenia rozmytych pojęć, takich

jak np. "podeszły wiek".

Maszyna wnioskująca realizuje wnioskowanie wstecz. Wewnętrzna reprezentacja reguł umożliwia, w razie potrzeby, wykorzystanie w tej samej bazie wiedzy również wnioskowania w przód (po odpowiedniej modyfikacji maszyny wnioskującej). Istniejący mechanizm objaśniania pozwala przejrzeć objaśnić uzyskane wnioski; brak natomiast mechanizmu, który udzielałby odpowiedzi na pytanie użytkownika: "Dlaczego o to pytasz?". Możliwe jest umieszczanie w treści reguł obliczeń arytmetycznych, warunków logicznych, komunikatów ostrzegawczych, i wykorzystywanie danych zawartych w dowolnych relacjach BD. Nie zakończono jednak realizacji środków umożliwiających pozyskiwanie wiedzy z informacyjnej bazy danych. Pozyskiwanie wiedzy jest jednak w pewnym stopniu wspierane wygodnym edytorem reguł, który posiada mechanizmy kontroli poprawności zapisu stwierdzeń, na podstawie znajomości jednego elementu stwierdzenia "podpowiada" pozostałe itp. Graficzny sposób określania stopnia pewności (por. 4.3.3.1 i ryc. 5.5, 6.10) ułatwia przekazanie przez inżyniera wiedzy lub użytkownika systemu jego intuicyjnej, subiektywnej oceny analizowanego faktu. Wykorzystanie środków SZBD do zrealizowania regułowej reprezentacji wiedzy umożliwia tworzenie bardzo dużej (do kilkudziesięciu tysięcy rozbudowanych reguł) bazy wiedzy. Aktualnie uzyskana efektywność czasowa jest zbyt niska, istnieją jednak możliwości jej poprawy (w czasie tworzenia programu, poza pewną dyscypliną programowania i stosowaniem wydajnych algorytmów, nie podejmowano specjalnych działań optymalizacyjnych). Przy okazji tych zmian należy starać się złagodzić, wynikającą z przyjętego rozwiązania niemożność użytkowania systemu doradczego w warunkach sieci komputerowej.

Porównanie powyższych uwag z zawartymi w punkcie 3.4 postulatami, pomimo wymienionych braków, wykazuje pomyślną realizację przyjętego projektu.

### 6.3. Ocena systemu informacyjnego.

System informacji o leku został zrealizowany jako relacyjna baza danych, o schemacie przedstawionym w punkcie 3.5.1. Zakres informacji obejmuje m.in.: dane fizyko-chemiczne, wskazania, przeciwwskazania, działania uboczne, interakcje, farmakokinetykę, dawkowanie, dostępność w kraju, nazwy zastrzeżone oraz piśmiennictwo. Pod względem zakresu uwzględnianych zagadnień oraz analizowanych źródeł informacji utworzony system jest prawdopodobnie najobszerniejszym systemem krajowym. Opracowanie dotyczy leków stosowanych w duszniczy bolesnej, zaburzeniach rytmu serca oraz nadciśnieniu tętniczym. Lekarze, którzy zapoznali się z działaniem systemu, pozytywnie ocenili dużą ilość (ponad 90) leków krążeniowych, a zwłaszcza obecność informacji o lekach niedawno wprowadzonych do lecznictwa.

Prezentacja wyników wyszukiwania jest przejrzysta, również formułowanie typowych, prostych pytań nie powinno być kłopotliwe. Jednak bardziej złożone pytania, wymagające określenia zbioru potrzebnych atrybutów lub zredagowania powiązań mogą sprawiać trudność początkującym użytkownikom.

W opinii autorów systemu niektóre informacje są zbyt zwięzłe - co jest spowodowane ograniczonymi możliwościami sprzętu komputerowego stosowanego w początkowej fazie pracy.

### 6.4. Ocena systemu doradczego.

Dla oceny systemu doradczego jego możliwości zostaną porównane z cechami idealnego SD, omawianymi w punkcie 2.2.

1°. Bezowocne poszukiwania. Starano się je wyeliminować przez wprowadzenie reguł klasyfikujących leki. Jednak wyczerpująca analiza prowadzona jest dla wszystkich leków mogących mieć zastosowanie w danym schorzeniu i jest ona kontynuowana także dla leków, dla których stwierdzono już zdecydowane przeciwwskazania. Innymi słowy system udziela wyczerpującej odpowiedzi nie tylko na zadane pyta-



nie, np. "Co jest wskazanym lekiem hipotensyjnym?", ale odpowiada też co jest lekiem hipotensyjnym niewskazanym.

2\*. Jakość rozwiązań. Testy systemu doradczego polegały na porównaniu propozycji systemu z terapią stosowaną przez lekarzy. Uczestniczyli w nich lekarze Kliniki Geriatrii AM w Krakowie oraz oddziału kardiologicznego Krakowskiego Szpitala Zespolonego im. G. Narutowicza. Przeanalizowano kilkanaście przypadków. Szczegółowa analiza przypadków 6 pacjentów Kliniki Geriatrii jest przedstawiona w pracy [23]. Wykazuje ona, że propozycje systemu były albo zgodne z postępowaniem lekarzy posiadających pewne doświadczenie w leczeniu chorób układu krążenia, albo były przez nich ocenione jako wariant możliwy do przyjęcia. Nigdy nie zdarzyło się, aby system zaproponował lek uznany przez lekarzy za przeciwwskazany.

3\*. Wnioskowanie symboliczne. Reprezentacja regułowa okazała się wystarczająca dla konstrukcji bazy wiedzy o zasadach stosowania leków krążeniowych.

4\*. Wiedza ogólna. System posiada niewielki zasób ogólnej wiedzy farmakologicznej, ograniczony do klasyfikacji leków wg grup farmakologicznych i zasad ich stosowania. Zna on także normy dla poziomu cholesterolu, kreatyniny, potasu itp. i potrafi np. ocenić stopień niewydolności nerek na podstawie poziomu kreatyniny. "Zdrowy rozsądek" ograniczono do kilku reguł głoszących np., że mężczyźni nie mogą być w ciąży, a kobiety i młodzi mężczyźni nie cierpią na przerost gruczołu krokowego.

5\*. Wyrażanie problemu w języku naturalnym. Oczywiście zastosowana składnia stwierdzeń daleka jest od bogactwa języka polskiego. Jednak konstruując bazę wiedzy udało się na ogół zachować poprawność gramatyczną budowanych stwierdzeń. Niechlubnymi wyjątkami są zdania postaci: "Zaburzenia rytmu serca jest stwierdzonym schorzeniem". W czasie wnioskowania system posługuje się wewnętrzną reprezentacją stwierdzeń (por. 3.4.2. i 3.4.3).

6°. Rozumowanie w warunkach niepewnej wiedzy. Przyjęte zasady rozumowania przybliżonego (zob. 2.5) dobrze sprawdziły się w działaniu. W ocenie lekarzy pojęcia takie jak podeszły wiek, czy niewydolność nerek powinny być "bardziej rozmyte".

7°. Rozwiązywanie trudnych problemów, ze złożonej dziedziny. W czasie testów odniesiono wrażenie, że system lepiej radzi sobie z trudniejszymi przypadkami. Wynika to zapewne stąd, że w skrajnie prostych przypadkach baza wiedzy dysponuje niewielką liczbą czynników różnicujących poszczególne leki, a pracochłonne wykluczanie nieistotnych czynników spowalnia działanie programu.

8°. Umiejętność wyjaśniania. System pozwala na pełne wyjaśnienie uzyskanych rezultatów, poprzez prezentację komunikatów objaśniających (komentarzy do reguł) oraz analizę przesłanek tych reguł. Możliwa jest oddzielna analiza wskazań i przeciwwskazań danego leku. Pewną trudność w ujawnianiu wskazań i przeciwwskazań konkretnego leku, stanowi fakt, że lek oceniany jest zarówno indywidualnie, jak i jako element grupy farmakologicznej. W takiej sytuacji, trzeba zapoznać się także z objaśnieniami odnoszącymi się do tej grupy. Niektóre reguły zawierają wniośki dotyczące leków należących do różnych grup. Ponieważ komentarz objaśniający odnosi się do całej reguły, część tekstu objaśnień może dotyczyć innych zagadnień, niż aktualnie istotne. Zwrócili na to uwagę lekarze, wskazując na potrzebę poprawy selektywności objaśnień. Można ją osiągnąć przez dekompozycję niektórych reguł obecnej bazy wiedzy.

9°. Możliwość rozumowania na temat własnej wiedzy. Jest ona ograniczona do ustalenia, czy przypadek nadaje się do analizy przez system. Np., w sytuacji, gdy pacjent jest dzieckiem, lub gdy farmakoterapia jest mało skuteczna, jak w przypadku wskazań do wszczepienia rozrusznika serca, system wyświetla komunikat: "Przypadek nie jest uwzględniony".

## 6.5. Wnioski.

Rezultatem pracy jest system spełniający większość postulowanych założeń. Ważniejsze jednak od powstałych, fizycznych struktur bazy danych, bazy wiedzy oraz zarządzającego nimi programu wydaje się uzyskane potwierdzenie celowości przedstawionej koncepcji systemu i zastosowanych technologii informatycznych, a także fakt usystematyzowania i wyrażenia w formalizmie reguł dostępnej wiedzy dotyczącej stosowania rozważanych leków. Doświadczenia nabyte w czasie realizacji systemu mogą być także przydatne, zarówno w dalszej pracy nad jego rozwojem, jak i przy tworzeniu innych systemów tego rodzaju. Poniżej zestawiono wnioski płynące z tych doświadczeń:

1\*. Przedstawienie informacji w postaci relacyjnej bazy danych pozwala szczegółowo ująć szereg zagadnień dotyczących leku. W stosunku do rozwiązania typu "system wyszukiwania dokumentów" umożliwia ono bardziej precyzyjne wyszukiwanie i daje możliwość efektywnego opracowania takich zagadnień jak np. interakcje leków. Nie wymaga konstrukcji tezaursu i żmudnej procedury indeksowania dokumentów.

2\*. Regułowa reprezentacja wiedzy, wbrew wątpliwościom wyrażonym w pracy Brodziaka i wsp.[22], okazała się odpowiednia do wyrażenia zasad farmakoterapii chorób układu krążenia. Dobrze sprawdziły się w działaniu przyjęte zasady rozumowania przybliżonego. Rozwiązania uzyskane przez komputer zostały zaakceptowane przez doświadczonych lekarzy. Technikę posługiwania się systemem ocenili oni jako łatwą do opanowania, a stawiane przezeń pytania jako celowe i jasne. Zasady wyrażania wiedzy w postaci reguł okazały się zrozumiałe dla osób bez wykształcenia informatycznego. Ułatwiło to współpracę z farmaceutą klinicznym w procesie pozyskiwania wiedzy.

3\*. Wyszukiwanie informacji przy pomocy środków dostarczonych przez SZBD, pomimo prostego w użyciu edytora zapytań, nastrocza jednak pewne trudności, zwłaszcza początkującym użytkownikom. Ponadto u-

zyskanie wszystkich potrzebnych informacji wymaga pewnej dodatkowej wiedzy.

Np. aby dowiedzieć się z jakimi lekami i jakie interakcje daje furosemid nie wystarczy zadać pytanie, wpisując w odpowiednią rubrykę nazwę tego leku. Trzeba również spytać o interakcje, które są charakterystyczne dla całej grupy diuretyków. Widać stąd, że dla udoskonalenia systemu informacji potrzebne jest wprowadzenie takiego modułu pośredniczącego pomiędzy informacyjną bazą danych a użytkownikiem, który ułatwiałby formułowanie pytań do systemu. Taką rolę mógłby spełniać prosty system doradczy, który byłby wyposażony w wiedzę zarówno o strukturze bazy danych, jak i z zakresu farmakologii. System oparty o bazę danych pozwala jedynie na odszukanie potrzebnych informacji, tak jak wyszukuje się je w książce (tyle, że wyszukiwanie jest dokładniejsze i znacznie szybsze). Dążąc do budowy systemu, który byłby także w stanie interpretować posiadaną wiedzę, należy zdecydować się na zastosowanie metod SI, np. reprezentacji wiedzy w postaci ram.

4\*. Wyniki teoretyczne dotyczące baz danych (szczególnie procesu normalizacji) mają ograniczone znaczenie w procesie konstrukcji informacyjnej bazy danych w tak złożonych dziedzinach, jak informacja o leku. Sprowadzanie relacji BD do coraz wyższych postaci normalnych wiąże się ze wzrostem ilości relacji tworzących bazę danych, co w rezultacie utrudnia użytkownikowi orientację w logicznej strukturze BD. Z drugiej strony, ze względu na zapewnienie przejrzystości tej struktury, zachodzi niekiedy potrzeba dekompozycji relacji, mimo że nadmiarowość nie występuje. W pracy starano się zachować kompromis pomiędzy nadmiarowością danych i logiczną złożonością systemu.

5\*. Technologia relacyjnych baz danych sprawdziła się natomiast jako wygodne narzędzie programisty. Oprócz podstawowego zastosowania, polegającego na opracowaniu informacji o leku i licznych mar-

ginesowych zastosowań (przechowywanie komunikatów i innych danych potrzebnych do pracy programu), relacje bazy danych zostały użyte do wewnętrznej reprezentacji regułowej bazy wiedzy, a procedury SZBD z powodzeniem wykorzystano do konstrukcji edytora bazy wiedzy i maszyny wnioskującej systemu doradczego. Zamiast tworzyć procedury do przetwarzania odpowiednich struktur w pamięci i ładowania ich z pamięci zewnętrznych wystarczyło zaprojektować pewien schemat bazy danych. Autor jest zdania, że w podobny sposób można zrealizować inne reprezentacje wiedzy, np. ramy. Dużą zaletą takiego rozwiązania jest znaczna redukcja pamięci operacyjnej wymaganej przez system, na rzecz pamięci zewnętrznej (dyskowej).

6°. Opracowany system może być wykorzystany do konstrukcji innych systemów informacyjno-doradczych dla innych grup leków, a także odpowiednich systemów z innych dziedzin; przy tym nie trzeba już ponownie ponosić nakładów związanych z konstrukcją programu (choć, oczywiście, dalsza praca nad istniejącym programem wydaje się konieczna). Systemy takie można z powodzeniem opracowywać w niewielkich zespołach. Z drugiej strony, program i fizyczna organizacja bazy danych czy bazy wiedzy są pod wieloma względami niedoskonałe; jednak zmiany w założeniach systemu nie wymagają ponownego wprowadzania danych czy reguł, gdyż są one już dostępne w pamięci komputera i mogą być w razie potrzeby automatycznie przekształcane do wymaganej postaci.

W przedstawionym rozwiązaniu zaledwie część dostępnej wiedzy o leku została wyrażona w jednej z reprezentacji oferowanych przez sztuczną inteligencję. Rzecz jasna dla niektórych danych, takich jak np. cena, nazwy zastrzeżone itp., umieszczenie ich w bazie danych jest najbardziej efektywne. Jednak wiele innych zagadnień, (np. mechanizm działania leku, przyczyny występowania interakcji itd.), obecnie zawartych w bazie danych w postaci tekstów, można by

skuteczniej wykorzystać po wyrażeniu ich np. w technice ram. Ni-  
niejsza praca stanowi zatem jedynie skromny krok w kierunku opraco-  
wania optymalnej koncepcji "inteligentnego" systemu informacji  
o leku.

## PRZYPISY

Wstęp.

- 1 Czasopismo "Informatyka" konsekwentnie używa określenia *system ekspertowy*, jako odpowiednika angielskiego terminu *expert system*. Używany przez autorów skryptu [7] termin *system doradczy* nie oddaje w pełni istoty tego pojęcia (jako, że nie każdy doradca bywa ekspertem), jednak po polsku brzmi dużo lepiej i będzie preferowany w tej pracy.

Rozdział 1.

- 1 Redundancja występuje wtedy, gdy pewna informacja przechowywana jest w kilku miejscach; niewielka redundancja jest na ogół korzystna, gdyż w razie utraty informacji daje szansę jej odzyskania, nadmierna redundancja powoduje nieefektywne wykorzystanie pamięci i wprowadza szereg innych trudności.
- 2 Kompresja danych ma na celu zmniejszenie pamięci zajmowanej przez te dane, na drodze odpowiedniego ich zakodowania.
- 3 Relacją na zbiorach  $A_1, A_2, \dots, A_n$  nazywamy dowolny podzbiór iloczynu kartezjańskiego tych zbiorów  $A_1 \times A_2 \times \dots \times A_n$ .
- 4 Jeżeli żaden podzbiór właściwy atrybutów nie jest kluczem to zbiór wszystkich atrybutów relacji jest kluczem (gdyż w relacji nie ma dwu identycznych krotek)
- 5 Problem nie wydaje się nowy, ale w dostępnym piśmiennictwie nie znaleziono rozważań na ten temat.

Rozdział 2.

- 1 J. von Neumann zaproponował aby pracą jednostki centralnej komputera sterował zakodowany ciąg rozkazów przechowywany w pamięci operacyjnej. Programowanie komputera polega więc, przynajmniej na poziomie języka wewnętrznego na zbudowaniu ciągu poleceń, czyli przedstawieniu algorytmu rozwiązywania zadania. Niemal wszystkie komputery, zarówno w skali mikro- jak i super- są budowane w oparciu o tę koncepcję.

### Rozdział 3.

- ' Nitroglicerynę pod nazwą NITROGLYCERIN produkują: STADAPHARM, D; RATIOPHARM, D; WANDER, CH; STREULI, CH i KABI, S. [10]
- ≈ HOECHST używa nazwy OSIREN zarówno dla leku o nazwie międzynarodowej SPIRONOLACTONE jak i CANRENOATE POTASSIUM [10].
- ≈ CIBA produkuje dihydralazyne pod nazwą NEPRESOL zarówno we Francji, jak i w Szwajcarii [10].

### Rozdział 4.

- ' Notacja wprowadzona przez polskiego logika, J. Łukasiewicza. Pozwala na zapis wyrażeń arytmetycznych bez użycia nawiasów. Spośród dwu wariantów tej notacji, częściej wykorzystywana jest (np. przy budowie kompilatorów [81]) notacja odwrotna. Jednak do przechowywania wyrażeń wygodniejsza jest notacja prosta.
- ≈ Rozszerzona notacja Backusa-Naura (Extended Backus-Naur Form) [63, 64].



## PIŚMIENICTWO

1. Date C. J.: Wprowadzenie do baz danych. WNT, W-wa 1981.
2. Martin J.: Organizacja baz danych. PWN, W-wa 1983.
3. Codd E. F.: A relational model of data for large shared data banks. CACM, 13, 6 (1970).
4. Codd E. F.: Further normalization of the data base relational model. [W:] Data Base Systems. Courant Computer Science Symposia Series. Vol. 6. Prentice-Hall, NY 1972.
5. Minsky M. L.: Artificial Intelligence. Scientific American, 215, 3 (1966).
6. Hayes-Roth F., Waterman D. A., Lenat D. B. (red.): Building Expert Systems. Addison-Wesley, Reading, Mass. 1983.
7. Cholewa W., Pedrycz W.: Systemy doradcze. Skrypt. Politechnika Śląska, Gliwice 1987.
8. Brandys J. i wsp.: Wprowadzenie do naukowej informacji o leku. Skrypt. Akademia Medyczna, Kraków 1988.
9. Lambert S., Ropiequet S. (red.): CD ROM. The New Papyrus. Microsoft Press, 1986.
10. Podlewski J. K., Chwalibogowska-Podlowska A.: Lekii współczesnej terapii. PZWL, W-wa 1985.
11. Hayes-Roth F.: The Knowledge-Based Expert System: A Tutorial. Computer, 9 (1984)
12. Muc M., Brodziak A.: Komputerowe wspomaganie lekarza w doborze środków farmakologicznych. Pol. Tyg. Lek. 40, 701 (1985).
13. Buller A., Birn P.: Komputerowe wspomaganie doboru środków farmakologicznych. Krajowa konferencja naukowa "Komputery w medycynie". Łódź, 1989.
14. Informacje własne.

15. Braczkowski R.: Własny system komputerowy wspomagający długoterminowe leczenie chorych na nadciśnienie tętnicze. Praca doktorska, Śl. AM, Katowice 1988.
16. Tokarski M.: Koncepcja systemu optymalnego wyboru leku oparta na teorii zbiorów rozmytych. Probl. Techn. Med. 13, 109 (1982)
17. Bogdanik T. i wsp.: Główne założenia programu komputerowego do leczenia nadciśnienia tętniczego. Pol. Tyg. Lek. 34, 1069 (1979)
18. Warmus M. i wsp.: Modelowanie matematyczne w przewlekłej zastoinowej niewydolności krążenia. Prace IPI PAN, 577. W-wa 1986.
19. Bobowska M. i wsp.: Mały Poradnik Terapeutyczny. PZWL, W-wa 1977.
20. Praca zbiorowa. Vademecum «Polfa». PZWL, W-wa 1981.
21. Muc M., Okoń Z., Brodziak A.: Nowe leki. Suplement poradników terapeutycznych. Skrypt Śl. AM, Katowice 1985.
22. Brodziak A. i wsp.: Jak zbudować samemu autorski, medyczny system ekspertowy. Materiały na XV sympozjum sekcji cybernetyki Towarzystwa Internistów Polskich. Bydgoszcz 1986.
23. Wójcik-Jawień M.: Baza wiedzy o lekach krążeniowych dla potrzeb konstrukcji komputerowego systemu informacji i systemu doradczego. Praca doktorska. AM Kraków, 1989.
24. Muc M., Brodziak A.: Analiza metod klasyfikacji środków farmaceutycznych w aspekcie możliwości wykorzystania ich podczas organizacji zbiorów danych o lekach w pamięci komputera. Probl. Tech. Med. 18, 4, 240-247 (1987)
25. Дрибас В. П.: Реляционные модели баз данных. Издательство БГУ им. В. И. Ленина, Минск 1982.
26. Cellary W., Królikowski Z.: Wprowadzenie do projektowania baz danych dBase III. WNT, W-wa 1988.

27. Codd E.F.: Relational completeness of data base sublanguages. [W:] Data Base Systems. Courant Computer Science Symposia Series. Vol. 6. Prentice-Hall, NY 1972.
28. Codd E.F.: Extending the Database Relational Model to Capture More Meaning. ACM Trans. on Database Syst. 4, 4, 397-434 (1979).
29. Codd E.F.: A data base sublanguage founded on the relational calculus. Proc. 1971 ACM SIGFIDET Workshop on Data Description, Access and Control.
30. Zloof M.M.: Query by example: a data base language. IBM Systems J., 16, 4 (1977)
31. Newell A., Simon H.A.: GPS - a program that simulates human thought. [W:] Feigenbaum E.A., Feldman J. (red.): Computer and Thought. McGraw-Hill, NY 1963.
32. Shortliffe E.H.: Computer-Based Medical Consultations MYCIN. American Elsevier, NY 1976.
33. Buchanan B.G., Feigenbaum E.A.: Dendral and Meta-Dendral. Their Application Dimension. AI 11, 5-24 (1978).
34. Duda R.O.: Model design in PROSPECTOR consultant system for mineral exploration. [W:] Michie D. (red.): Expert systems in the micro-electronic age. Edinburgh Univ. Press, s. 153-167.
35. Kunz J.C. 1 wsp.: A physiological rule-based system for interpreting pulmonary function test results. Heuristic Programming Project, Report HPP-78-19. Stanford University, Stanford 1978.
36. Miller R.A. 1 wsp.: INTERNIST-I, an experimental computer-based diagnostic consultant for general internal medicine. N. Engl. J. Med. 1307, 468, 1982.
37. Abbot Pharmacokinetic Systems Theophylline Program Operations Manual. Abbot Laboratories, Chicago, 1985.

38. Sheiner L.B. i wsp.: Forecasting individual pharmacokinetics. Clin. Pharm. Ther. 26, 3, 294-305 (1979)
39. Peck C.C.: Computer-assisted Clinical Pharmacokinetics. [W:] Bennet L.Z., Massoud N., Gambertoglio J.G. (red.): Pharmacokinetic Basis for Drug Treatment. Raven Press, NY 1983.
40. Grzegorzczak A.: Zarys logiki matematycznej. PWN, W-wa 1984.
41. Kluźniak F., Szpakowicz S.: Prolog. WNT, W-wa 1983.
42. Clark K.L., McCabe F.G.: Micro-Prolog. WNT, W-wa 1988.
43. Hayes-Roth F.: Rule-Based Systems. CACM, 28, 9, 921-932 (1985)
44. Bolc L., Cichy M., Różańska M.: Przetwarzanie języka naturalnego. WNT, W-wa 1982.
45. Fikes R., Kehler T.: The Role of Frame-Based Representation in Reasoning. CACM, 28, 9, 904-920 (1985).
46. Vozeh S. i wsp.: Rapid Prediction of Steady-State Serum Theophylline Concentration in Patients Treated with Intravenous Aminophylline. Eur. J. Clin. Pharmacol. 18, 473-477 (1980).
47. Zadeh L.A.: Fuzzy sets. Inform. Control. 8, 6, 338-353 (1965).
48. Czogała E., Pedrycz W.: Elementy i metody <sup>teorii</sup> zbiorów rozmytych. PWN, W-wa 1985.
49. Shortliffe E.H., Buchanan B.G.: A Model of Inexact Reasoning in Medicine. Math. Biosci. 23, 351-379 (1975)
50. Abramowicz W.: Modularny system wyszukiwawczy. Informatyka 21, 1, 6-10 (1986).
51. Kent W.: Limitations of Record-Based Information Models. ACM Trans. on Database Syst. 4, 1, 107-131 (1979).
52. Chen P.P.S.: The entity-relationship model toward a unified view of data. ACM Trans. on Database Syst. 1, 1 (1976)
53. Gallaire H. i wsp.: Logic and Databases: A Deductive Approach. ACM Comp. Surv. 16, 2, 153-185 (1984).

54. Kirpluk M., Sobolewski P.: Implementacja dedukcyjnej bazy danych Holmes. Informatyka 23, 2, 20-23 (1988)
55. Roach J., Saiyuen L.: An expert system for information on pharmacology and drug interactions. Comput. Biol. Med., 15, 1, 11-23 (1985)
56. Delobel C., Casey R.G.: Decomposition of a Data Base and the Theory of Boolean Switching Functions. IBM J. Res. Develop. 17, 9, 374-386 (1973)
57. Fagin R.: Functional Dependencies in a Relational Database and Propositional Logic. IBM J. Res. Develop. 21, 11, 534-544 (1977)
58. Łukaszewicz W.: Przechowywanie wiedzy i niewiedzy w bazach danych. Informatyka 21, 2, 1-3 (1987).
59. Wirth N.: Algorytmy + struktury danych = programy. WNT, W-wa 1980.
60. Kruglinsky D.: Data Base Management Systems. A Guide to Microcomputer Software. Osborne/McGraw-Hill, Berkeley 1983.
61. Jagielski R.: Tablice rozproszone. WNT, W-wa 1982.
62. Fagin R. i wsp.: Extendible Hashing - A Fast Access Method for Dynamic Files. ACM Trans. on Database Syst. 4, 3, 315-344 (1979).
63. Jensen K., Wirth N.: Pascal user manual and report. Lecture Notes in Computer Science. 18, Springer-Verlag, NY 1976. (wyd. 2 popr.)
64. Iglewski M., Madey J., Matwin S.: Pascal. WNT, W-wa 1986.
65. Kott R.K.: Programowanie w języku Pascal. WNT, W-wa 1988.
66. Oprogramowanie minikomputera MERA-60. Opis języka Pascal-60. CNPSS «MERA-STER», Katowice 1983.
67. Turbo Pascal version 3.0 Reference Manual. Borland International, Scotts Valley 1985.
68. Turbo Pascal version 4.0 Reference Manual. Borland Interna-

- tional, Scotts Valley 1987.
69. Kernighan B.W., Ritchie D.M.: Język C. WNT, W-wa 1987.
70. Maurer W.D.: A Programmer's Introduction to LISP. Macdonald/American Elsevier, London-NY 1971.
71. dBase III. User manual. Ashton-Tate, 1984.
72. EXSYS. Expert System Development Package. Exsys Inc., 1985.
73. Sato S., Sugimoto M.: Artificial Intelligence. Fujitsu Sci. Tech. J. 22, 3, 139-181 (1986).
74. Jawień W.: Konstrukcja translatora rozszerzalnego języka programowania opartego o LISP. Praca magisterska. UJ, Kraków 1981.
75. Didising D.: Cognitive Software Designs Using Warnier. Proc. IFIP-IMIA International Conf. Computer-Aided Medical Decision Making, Praga 1985.
76. Boyer R.S., Moore J.S.: A Fast String Searching Algorithm. CACM 20, 10, 762-772 (1977).
77. Jarke M., Koch J.: Query Optimization in Database Systems. ACM Comp. Surv. 16, 2, 111-152 (1984).
78. Gries D.: Konstrukcja translatorów dla maszyn cyfrowych. PWN, W-wa 1984.
79. Brown P.J.: Makrogeneratory i oprogramowanie przenośne. WNT, W-wa, 1978.
80. Hopgood F.R.A.: Metody kompilacji. PWN, W-wa 1982.
81. Rohl J.S.: An Introduction to Compiler Writing. McDonald/American Elsevier, London-NY 1971.
82. Foster J.M.: Automatyczna analiza składniowa. PWN, W-wa 1976.

