METODY DOWODZENIA TWIERDZEŃ I AUTOMATYZACJA ROZUMOWAŃ
KONWERSATORIA 2015/2016

V rok kognitywistyki UAM

1 Uwagi organizacyjne

- Zajęcia 1–8: Jerzy Pogonowski (obie grupy)
- Zajęcia 9-15: Szymon Chlebowski (obie grupy)

1.1 Spis tematów konwersatorium

1. Powtórka: podstawowe pojęcia logiczne. Pełne drzewo dwójkowe. Notacja Smullyana. Algebra Lindenbauma-Tarskiego.
2. Zadania: operacje konsekwencji, indukcja strukturalna. Reprezentacja reguł wnioskowania w zbiorze skończonych ciągów dwójkowych.
3. Zadania: metoda aksjomatyczna
4. Zadania: postacie normalne i prefiksowe
5. Zadania: tablice Smullyana
6. Zadania: rezolucja
7. Zadania: dual tableaux (diagramy Rasiowej-Sikorskiego)
8. Kolokwium I
9. Zadania: dedukcja naturalna
10. Zadania: rachunek sekwentów
11. Zadania: Haskell
12. Zadania: Haskell
13. Zadania: Haskell

14. Zadania: Haskell

15. Kolokwium II.

1.2 Zasady zaliczania

1. Punktacja: uzyskać możesz maksymalnie 70 punktów. Kolokwium=30 punktów, aktywność na zajęciach=10 punktów. Zadania domowe wchodzą w skład aktywności.

2. Przykładowe zadania Kolokwium I:

   (a) Korzystając z twierdzenia o dedukcji zbuduj dowód podanej formuły w systemie aksjomatycznym.

   (b) Sprowadź do postaci normalnej podaną formułę.

   (c) Zbuduj dowód tablicowy podanej formuły.

   (d) Zbuduj dowód rezolucyjny podanej formuły.

3. Przykładowe zadania Kolokwium II:

   (a) Zbuduj dowód podanej formuły w systemie dedukcji naturalnej.

   (b) Zbuduj dowód podanej formuły w rachunku sekwentów.

   (c) Zadania dotyczące automatyzacji rozumowań w języku Haskell poda Pan mgr Szymon Chlebowski.

1.3 Literatura

1. Fitting, M. 1996. *First-Order Logic and Automated Theorem Proving*. Springer, Berlin.

2. Ławrow, I.A., Maksimowa, Ł.L. 2004. *Zadania z teorii mnogości, logiki matematycznej i teorii algorytmów*. Wydawnictwo Naukowe PWN, Warszawa.

3. Pogorzelski, W.A. 1992. *Elementarny słownik logiki formalnej*. Wydawnictwo Filii Uniwersytetu Warszawskiego, Białystok.

4. *Stanford Encyclopedia of Philosophy*: http://plato.stanford.edu/ (artykuły poświęcone teorii dowodu oraz automatyzacji rozumowań: *The development of proof theory*, *Automated reasoning*).

Dodatkowa literatura będzie polecana w trakcie zajęć. Proponuję także lekturę zamieszczonych na stronach ZLiK materiałów dydaktycznych: Pani Dr Doroty Leszczyńskiej-Jasion, Pana Prof. Mariusza Urbańskiego oraz Pana prof. Andrzeja Wiśniewskiego.

1.4 Materiały dydaktyczne

Będą dostępne na stronie: http://logic.amu.edu.pl/index.php/Dydaktyka

Tam również: syllabus.

1.5 Założenia o słuchaczach

Korzystać będziemy z wiadomości przekazanych na kursach:

- *Wprowadzenie do logiki*
- *Logika I*
- *Logika II*

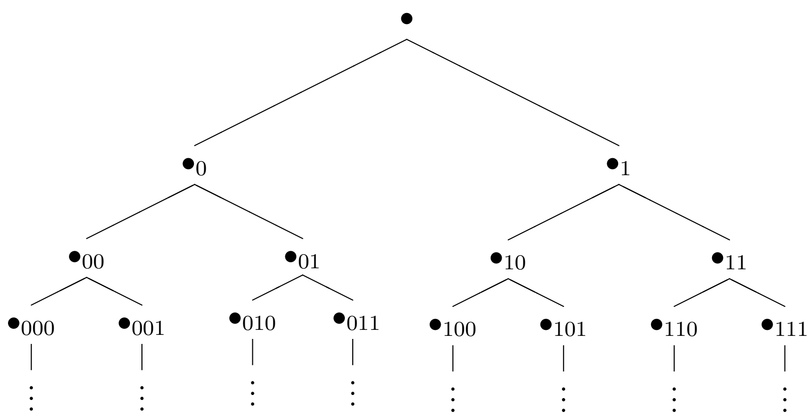
Kurs *Matematyczne podstawy kognitywistyki* nie był obowiązkowy dla studentów obecnego V roku. Potrzebne nam pojęcia matematyczne będą omawiane na bieżąco.

2 Powtórka

2.1 KRZ

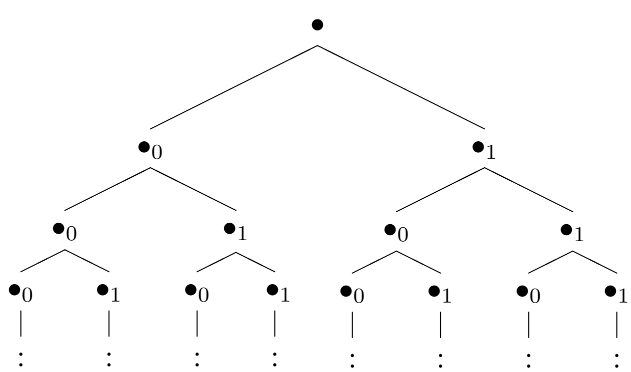
2.1.1 Pełne drzewo dwójkowe

Pamiętamy, że wartościowania w KRZ są nieskończonymi ciągami o wyrazach 0 lub 1. Wszystkie wartościowania reprezentować można w postaci *pełnego drzewa dwójkowego*:



Każdy z kolejnych wierzchołków ma dwóch bezpośrednich potomków. Wierzchołki (oprócz korzenia) kodujemy ciągami zer i jedynek. Tak więc, jeśli jakiś wierzchołek ma kod σ, to jego bezpośrednimi potomkami są wierzchołki o kodach: $\sigma 0$ oraz $\sigma 1$. *Gałęzią* nazwiemy każdy *nieskończony* ciąg złożony z zer i jedynek.

Możemy też patrzeć na pełne drzewo dwójkowe w sposób następujący. Każdy wierzchołek ma dwóch bezpośrednich potomków: lewego potomka znakujemy przez 0, prawego przez 1. Ta reprezentacja pełnego drzewa dwójkowego wygląda zatem następująco:



Ćwiczenie. Wykorzystując tę drugą reprezentację pokaż, że nie jest możliwe ponumerowanie (liczbami naturalnymi: 0, 1, 2, 3, 4, 5,...) wszystkich gałęzi pełnego drzewa dwójkowego, czyli wszystkich nieskończonych ciągów o wyrazach 0 lub 1.

Rozwiązanie wykorzystuje *metodę przekątniową* Cantora. Przypuśćmy, dla dowodu nie wprost, że można wszystkie gałęzie nieskończonego drzewa dwójkowego ponumerować liczbami naturalnymi. Niech to wyliczenie ma postać następującą (każda a_i^j jest zerem lub jedynką)::

1. $g_1 = a_1^1 a_1^2 a_1^3 \ldots$
2. $g_2 = a_2^1 a_2^2 a_2^3 \ldots$
3. $g_3 = a_3^1 a_3^2 a_3^3 \ldots$
4. itd.

Rozważmy ciąg $G = b_1 b_2 b_3 \ldots$, gdzie:

1. jeśli $a_n^n = 0$, to $b_n = 1$
2. jeśli $a_n^n = 1$, to $b_n = 0$.

Wtedy ciąg G różni się od *każdego* z ciągów g_n (co najmniej na n-tym miejscu). Tak więc, jakkolwiek chcielibyśmy ponumerować wszystkie gałęzie pełnego drzewa dwójkowego liczbami naturalnymi, to zawsze pozostaną gałęzie, dla których numerów nie starczy.

Zauważmy, że nasze przypuszczenie dotyczyło *dowolnego* sposobu numerowania wszystkich gałęzi drzewa dwójkowego liczbami naturalnymi. Powyższy wynik oznacza zatem, że taka (wyczerpująca wszystkie gałęzie) numeracja jest niemożliwa. Tak więc wszystkich gałęzi tego drzewa nie można ustawić w ciąg uporządkowany tak, jak wszystkie liczby naturalne.

2.1.2 Pełna tabela funktorów

Wszystkie dwuargumentowe funkcje prawdziwościowe:

p	q	$\perp$	$\wedge$	$\nrightarrow$	p	$\nleftarrow$	q	$\not\equiv$	$\vee$	$\downarrow$	$\equiv$	$\neg q$	$\leftarrow$	$\neg p$	$\rightarrow$	$\uparrow$	$\top$
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Wszystkie funkcje 1-argumentowe:

arg	1	2	3	4
0	0	0	1	1
1	0	1	0	1

2.1.3 Notacja Smullyana

We współczesnych podręcznikach coraz bardziej popularna staje się konwencja notacyjna zaproponowana przez Smullyana, która pozwala w skrótowej formie zapisywać typy formuł oraz reguły dowodowe.

Po pierwsze, podzielmy funktory języka KRZ na:

1. *pierwszorzędne*: koniunkcja, alternatywa (nierozłączna), implikacja prosta, implikacja odwrotna, kreska Sheffera (zaprzeczenie koniunkcji), binegacja (zaprzeczenie alternatywy), zaprzeczenie implikacji prostej oraz zaprzeczenie implikacji odwrotnej;

2. *drugorzędne*: pozostałe funktory dwuargumentowe (a więc także równoważność oraz jej zaprzeczenie, czyli alternatywę wykluczającą, a również cztery funktory, które równoważne są pierwszemu argumentowi, jego zaprzeczeniu, drugiemu argumentowi, jego zaprzeczeniu, a wreszcie *verum* $\top$ oraz *falsum* $\bot$).

Funktor $\uparrow$ jest (przez informatyków) nazywany NAND, natomiast $\downarrow$ nazywany jest (przez informatyków) NOR.

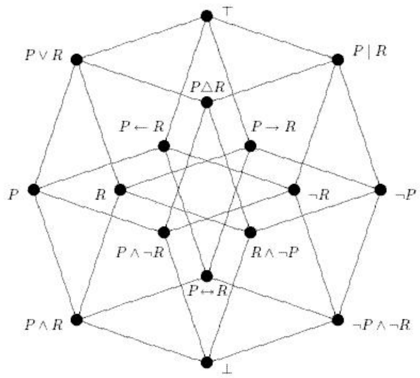
Po drugie, wśród funktorów pierwszorzędnych oraz ich zaprzeczeń wyróżnimy te, które „działają" koniunkcyjnie oraz te, które „działają" alternatywnie. Formuły z tymi pierwszymi funktorami oznacza się symbolem α, te drugie zaś symbolem β. *Składniki* takich formuł są oznaczane symbolami, odpowiednio: α_1, α_2 oraz β_1, β_2. Składniki te wyznaczane są wedle następującej konwencji:

α	α_1	α_2
$\varphi \wedge \psi$	φ	ψ
$\neg(\varphi \vee \psi)$	$\neg\varphi$	$\neg\psi$
$\neg(\varphi \to \psi)$	φ	$\neg\psi$
$\neg(\varphi \leftarrow \psi)$	$\neg\varphi$	ψ
$\neg(\varphi \uparrow \psi)$	φ	ψ
$\varphi \downarrow \psi$	$\neg\varphi$	$\neg\psi$
$\varphi \nrightarrow \psi$	φ	$\neg\psi$
$\varphi \nleftarrow \psi$	$\neg\varphi$	ψ

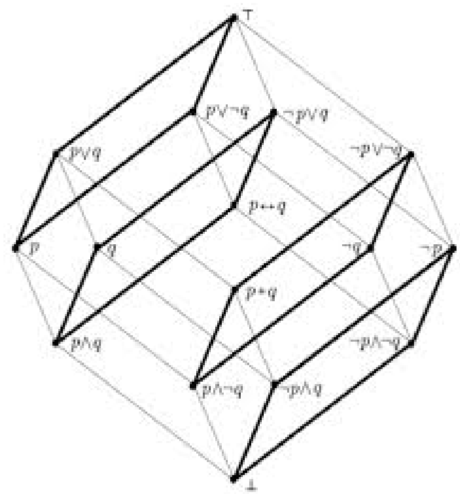
β	β_1	β_2
$\neg(\varphi \wedge \psi)$	$\neg\varphi$	$\neg\psi$
$\varphi \vee \psi$	φ	ψ
$\varphi \to \psi$	$\neg\varphi$	ψ
$\varphi \leftarrow \psi$	φ	$\neg\psi$
$\varphi \uparrow \psi$	$\neg\varphi$	$\neg\psi$
$\neg(\varphi \downarrow \psi)$	φ	ψ
$\neg(\varphi \nrightarrow \psi)$	$\neg\varphi$	ψ
$\neg(\varphi \nleftarrow \psi)$	φ	$\neg\psi$

2.1.4 Algebra Lindenbauma-Tarskiego

Przedstawimy na diagramie szesnaście formuł języka KRZ z poszczególnymi funktorami prawdziwościowymi. Najpierw diagramy dostępne w sieci:



The Lindenbaum–Tarski algebra of
a classical propositional language generated
by the two variables $\{P, R\}$.



Teraz narysujemy samodzielnie stosowny diagram, który – naszym zdaniem – przedstawia omawianą algebrę w sposób ułatwiający m.in. zapamiętanie notacji Smullyana. Najpierw zastanów się, jak przedstawić uporządkowanie, ze względu na inkluzję, rodziny wszystkich podzbiorów zbioru o czterech elementach. Potem zastanów się, jak na płaszczyźnie przedstawić *tesserakt* (hipersześcian, sześcian czterowymiarowy). Wreszcie, narysuj diagram Hassego rozważanej algebry Lindenbauma-Tarskiego. Patrzymy na tablicę. Rysujemy wspólnie: ja wprowadzam niektóre wierzchołki, studenci ustalają, gdzie powinny leżeć pozostałe wierzchołki.

RYSUJEMY NA TABLICY

Ten rysunek powinien ułatwić zapamiętanie m.in.:

1. które spójniki są pierwszorzędne (atomy i koatomy)

2. które formuły są koniunkcyjne (atomy), a które alternatywne (koatomy)

3. jakie związki koniunkcyjno-alternatywne zachodzą pomiędzy tymi formułami (kresy w algebrze, wyznaczone przez $\wedge$ oraz $\vee$).

Diagram ten przedstawia algebrę Lindenbauma-Tarskiego (dla formuł o dwóch zmiennych). Jak widzieliśmy podczas jego rysowania, odpowiada on także algebrze Boole'a wszystkich podzbiorów zbioru czteroelementowego.

Oglądamy również mój wspaniały tekturowy *dwunastościan rombowy*, który także wiąże się z omawianą strukturą.

Problemy. Jeśli mamy formułę o dwóch zmiennych wolnych, ale naprawdę bardzo długą – miliardy razy występują w niej koniunkcje, alternatywy, negacje, implikacje, to gdzie znajdzie się ona na powyższym diagramie? Ile elementów ma algebra Lindenbauma-Tarskiego dla formuł z trzema zmiennymi?

2.2 KRP

2.2.1 Notacja Smullyana

Notacja Smullyana dla języków pierwszego rzędu (jak język KRP) oprócz powyższych konwencji dla funktorów prawdziwościowych uwzględnia jeszcze notację dla formuł skwantyfikowanych oraz ich negacji. Rozróżnia się dwa typy: γ-*formuły*, które „działają" uniwersalnie (czyli formuły z kwantyfikatorem generalnym lub z zaprzeczeniem kwantyfikatora egzystencjalnego) oraz δ-*formuły*, które „działają" egzystencjalnie (czyli formuły z kwantyfikatorem egzystencjalnym lub z zaprzeczeniem kwantyfikatora generalnego). Dla każdego z tych typów formuł oraz dowolnego termu t określa się ich *instancje* w sposób następujący:

γ	$\gamma(t)$	δ	$\delta(t)$
$\forall x\varphi$	$\varphi(x/t)$	$\exists x\varphi$	$\varphi(x/t)$
$\neg\exists x\varphi$	$\neg\varphi(x/t)$	$\neg\forall x\varphi$	$\neg\varphi(x/t)$

Ta konwencja zostaje nieco zmodyfikowana w przypadku języków pierwszego rzędu z symbolami funkcyjnymi (gdzie uwzględniamy dodatkowo unifikację termu występującego w instancjach), jak zobaczymy później.

Wreszcie, notację Smullyana stosuje się także np. w językach, w których występują modalności. Mamy więc π-*formuły* (postaci $\Diamond\varphi$) oraz ν-*formuły* (postaci $\Box\varphi$). Stosowne reguły poznamy w późniejszych wykładach.

2.2.2 Metoda kontrprzykładu

Pamiętamy, że KRP jest nierozstrzygalny, a więc nie istnieje algorytm, który dla dowolnej formuły języka KRP rozstrzyga, czy jest ona tautologią KRP czy nie jest. Istnieją *półalgorytmy* (np. metoda tablic analitycznych), które dają odpowiedź pozytywną, gdy badana formuła *jest* tautologią, ale mogą nie dać odpowiedzi w przeciwnym przypadku.

W niektórych przypadkach nietrudno ustalić, że dane zdanie *nie jest* tautologią, podając przykład interpretacji, w której jest ono fałszywe.

Ćwiczenie. Pokaż, że nie są tautologiami:

1. $\forall x(P(x) \vee Q(x)) \rightarrow (\forall x P(x) \vee \forall x Q(x))$
2. $(\exists x P(x) \wedge \exists x Q(x)) \rightarrow \exists x(P(x) \wedge Q(x))$
3. $\forall x \exists y P(x, y) \rightarrow \exists y \forall x P(x, y)$

2.2.3 Rozumienie kwantyfikatorów

Pamiętamy, że definicja spełniania formuły w interpretacji przez wartościowanie odwołuje się, w przypadku formuł z kwantyfikatorami do innych wartościowań, spełniających formułę w zasięgu kwantyfikatora.

Można też rozważać *podstawieniową* interpretację kwantyfikatorów, przy której zakłada się, że wszystkie elementy uniwersum U interpretacji M mają nazwy w języku. Niech nazwą elementu $a \in U$ będzie $\overline{a}$. Wtedy warunki dla formuł z kwantyfikatorami w definicji spełniania formuły w interpretacji M przez wartościowanie w przybierają postać:

1. $M \models_w \forall x \varphi(x)$ wtedy i tylko wtedy, gdy $M \models_w \varphi(\overline{a})$ dla wszystkich $a \in U$
2. $M \models_w \exists x \varphi(x)$ wtedy i tylko wtedy, gdy $M \models_w \varphi(\overline{a})$ dla pewnego $a \in U$.

2.2.4 Nieskończoność

Zakładamy, że słuchacze pamiętają z kursu *Matematycznych podstaw kognitywistyki* definicję zbioru nieskończonego (w sensie Dedekinda): zbiór X jest *nieskończony*, gdy jest równoliczny ze swoim podzbiorem właściwym. Zbiory, które nie są nieskończone nazywamy *skończonymi*. Zakładamy też znajomość twierdzenia Cantora: żaden zbiór nie jest równoliczny z rodziną wszystkich swoich podzbiorów.

Zbiór nieskończony jest:

1. *przeliczalny*, gdy jest równoliczny ze zbiorem wszystkich liczb naturalnych;

2. *nieprzeliczalny*, gdy nie jest przeliczalny.

Jak widzieliśmy wyżej, zbiór wszystkich gałęzi pełnego drzewa dwójkowego (czyli także zbiór wszystkich wartościowań w KRZ) jest nieprzeliczalny.

Ćwiczenie. Podaj przykład zdania języka KRP, które prawdziwe jest jedynie w strukturach nieskończonych.

Ćwiczenie. Jeśli X jest dowolnym zbiorem, to podzbiorem *koskończonym* zbioru X nazywamy każdy zbiór $Y \subseteq X$ taki, że $X - Y$ jest skończony. Czy rodzina wszystkich koskończonych podzbiorów zbioru wszystkich liczb naturalnych $\mathbb{N}$ jest domknięta na operacje:

1. iloczynu
2. sumy
3. dopełnienia (względem X)?

Jakie jeszcze własności tej rodziny potrafisz wskazać?

3 Ciekawostki

3.1 Notacja Łukasiewicza

Przyzwyczajeni jesteśmy do notacji *infiksowej* dla funktorów dwuargumentowych (symbol funktora między symbolami argumentów). Potrzebne są wtedy nawiasy, albo inne znaki interpunkcji dla zaznaczania zasięgu funktorów. W notacji Łukasiewicza (zwanej też *notacją polską*) symbol funktora stawiamy przed symbolami jego argumentów. Oczywiście przyjąć trzeba ustalone symbole dla funktorów (różne od symboli dla zmiennych), np.:

1. N – negacja
2. A – alternatywa
3. K – koniunkcja
4. C – implikacja
5. E – równoważność.

Ćwiczenie. Dokonaj przekładu między notacją polską a infiksową:

1. $((p \to q) \wedge (p \to \neg q)) \to q$

2. $p \to (q \to (r \to (s \to t)))$

3. $(((p \to q) \to r) \to s) \to t$

4. $EApqAKpqAKpNqKNpq$

5. $CCCpqCCCNrNstrCuCCrpCsp$

3.2 Diagramy Venna i Carrolla

Zależności między zakresami nazw reprezentować można graficznie na wiele sposobów. Dla przykładu:

3.2.1 Diagramy Venna

Jakieś ładne figury wybieramy dla oznaczenia zakresów nazw – np. okręgi lub elipsy. Na diagramie Venna każde dwa zakresy zachodzą na siebie. Tak więc, dla n zbiorów mamy na diagramie Venna 2^n obszarów. Można umówić się, że obszary puste oznaczamy np. znakiem $-$, a obszary niepuste np. znakiem $+$.

Diagram Venna dla trzech zbiorów:

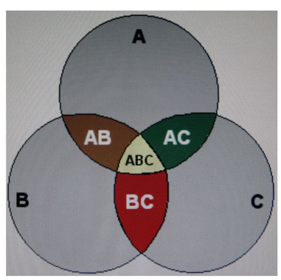
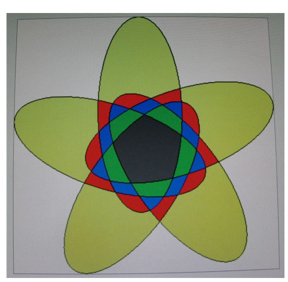


Diagram Venna dla pięciu zbiorów:



Jak można to wykorzystać? Na przykład:

1. W ocenie poprawności rozumowań reprezentowanych w monadycznym rachunku predykatów.

2. W ustalaniu, czy warunki podane dla branych pod uwagę zbiorów są niesprzeczne.

Przykład 1. Jesteś na intensywnej terapii. Trzeba ci **natychmiast** podać lek zawierający jednocześnie alfaminę, betaminę oraz deltaminę. [Nazwy leków są zmyślone, jak mi się wydaje. Nie jestem opłacany przez żadną firmę medyczną.] Pielęgniarce trzęsą się ręce i próbuje sobie przypomnieć:

Zaraz, jak to było... Ten stary łysy profesor coś tam o tym bredził, na tym wykładzie, podczas którego podrywałam Roberta... Każda alfamina jest też betaminą. Niektóre betaminy są deltaminami. Jeżeli lek jest betaminą lub deltaminą, to jest również alfaminą. Co prawda, nie ma leku, który jest alfaminą i betaminą, lecz nie jest deltaminą. Ale czy to wszystko oznacza, że jest lek, którego ona potrzebuje?! Joszua, Miriam!!! Dla niej nie ma ratunku!

Ona rozmyśla, czas płynie. **Twój** czas właśnie się **kończy**... Bo przecież nie ma dla ciebie ratunku, prawda? Przyjmijmy, że to, co mamrocze pielęgniarka ***jest prawdą***. Czy istnieje lek zawierający alfaminę, betaminę oraz deltaminę?

Przyjmiemy oznaczenia:

- $A(x)$ — x jest alfaminą;
- $B(x)$ — x jest betaminą;
- $D(x)$ — x jest deltaminą.

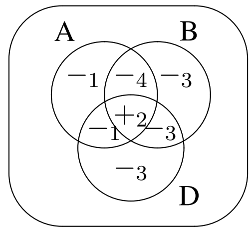
Wiadomości zapamiętane przez pielęgniarkę zapisane w języku KRP mają postać:

1. $\forall x\ (A(x) \to B(x))$
2. $\exists x\ (B(x) \land D(x))$
3. $\forall x\ ((B(x) \lor D(x)) \to A(x))$
4. $\neg\exists x\ ((A(x) \land B(x)) \land \neg D(x))$.

Najpierw pokażemy, że: a) wiadomości zapamiętane przez pielęgniarkę są semantycznie niesprzeczne.

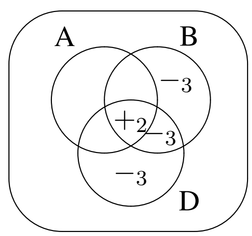
Potem zaś pokażemy, że: b) z 2. oraz 3. wynika logicznie dająca Ci ratunek formuła:

$$(\bigstar) \quad \exists x\ (A(x) \land (B(x) \land D(x))).$$



Z diagramu tego widać, że $A = B = D = A \cap B \cap D \neq \emptyset$.

Nadto, jeśli sporządzimy taki diagram tylko dla warunków 2. oraz 3., to zobaczymy, iż obszar $A \cap B \cap D$ jest niepusty (w ogólności trzeba jednak sprawdzić wszystkie warunki, aby wykluczyć, że omawiane warunki są sprzeczne):



Przeżyjesz, jeśli pielęgniarka zrobi szybki użytek z Logiki.

Przykład 2. Czy z poniższych przesłanek wynika jakiś wniosek dotyczący zależności między inteligentnymi a sympatycznymi? Ponadto: co można powiedzieć o uczciwych, którzy nie są sympatyczni?

> *Co najmniej jeden uczciwy jest sympatyczny. Nie wszyscy są uczciwi. Każdy jest uczciwy lub inteligentny lub sympatyczny. Wszyscy inteligentni są uczciwi lub sympatyczni. Wszyscy uczciwi inteligentni są sympatyczni. Wszyscy sympatyczni są uczciwi lub inteligentni. Żaden uczciwy sympatyczny nie jest inteligentny.*

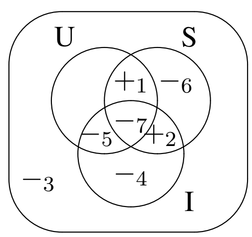
Wprowadźmy oznaczenia:

- $U(x)$ — x jest uczciwy
- $I(x)$ — x jest inteligentny

- $S(x)$ — x jest sympatyczny.

Rozważane przesłanki mają następujące schematy:

1. $\exists x\ (U(x) \wedge S(x))$
2. $\neg\forall x\ U(x)$
3. $\forall x\ (U(x) \vee (I(x) \vee S(x)))$
4. $\forall x\ (I(x) \rightarrow (U(x) \vee S(x)))$
5. $\forall x\ ((U(x) \wedge I(x)) \rightarrow S(x))$
6. $\forall x\ (S(x) \rightarrow (U(x) \vee I(x)))$
7. $\neg\exists x\ (U(x) \wedge (S(x) \wedge I(x)))$.



Z powyższego diagramu widać, że (przy prawdziwości przesłanek):

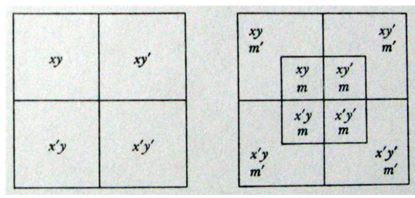
- *Istnieją inteligentni i sympatyczni. Wszyscy inteligentni są sympatyczni. Istnieją sympatyczni, którzy nie są inteligentni.*
- *Jeśli ktoś jest uczciwy, ale nie jest sympatyczny, to nie jest inteligentny. Nie wiadomo jednak, czy istnieją uczciwi, którzy nie są sympatyczni i nie są inteligentni.*

Problem: jaką notację zaproponujesz dla oznaczania niepustości sumy obszarów?

3.2.2 Diagramy Carrolla

Diagramy Carrolla dla dwóch i trzech zbiorów:

	B	$-B$
A	$A \cap B$	$A \cap -B$
$-A$	$-A \cap B$	$-A \cap -B$



Jak można to wykorzystać? Tak samo, jak diagramy Venna.

Jerzy Pogonowski
Zakład Logiki i Kognitywistyki UAM
www.kognitywistyka.amu.edu.pl
http://logic.amu.edu.pl/index.php/Dydaktyka
pogon@amu.edu.pl