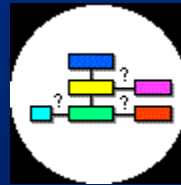


# Sztuczna Inteligencja

## 2. Szukanie - sformułowanie problemu.

W którym to zobaczymy, jak agent może znaleźć rozwiązanie w sytuacji, w której żadne proste działanie nie wystarczy.



Włodzisław Duch

Katedra Informatyki Stosowanej UMK

Google: W. Duch

# Szukanie

Szukanie to jedna z najważniejszych metod informatyki, niemal utożsamiana ze sztuczną inteligencją.

Występuje w wielu problemach:

- dedukcji, rozumowania, wnioskowania, planowania, dowodzenia, optymalizacji ...

Donald Knuth poświęcił szukaniu cały tom:

„The art of computer programming. Vol. III. Sorting & Search”.

Wikipedia: [lista algorytmów szukania](#).

## Szukajcie a (być może) znajdziecie!

Systematyczna eksploracja alternatyw.

Rezultat: sekwencja kroków prowadząca do rozwiązania.

# Gdzie szukać?

Jak zdefiniować przestrzeń poszukiwań?

Konieczna jest jakaś reprezentacja problemu.



Hipoteza „Przestrzeni Problemów” (Allen Newell):

szukanie w PP to ogólny model inteligentnego działania;

celowe działania symboliczne zachodzą w przestrzeni problemów.

Klasyczne problemy, które można w ten sposób rozwiązać:

- Przesuwanki, np. kostka Rubika;
- labirynty, poszukiwanie optymalnej drogi;
- problemy układania klocków, np. wieża z Hanoi;
- zagadki logiczne, np. misjonarze i kanibale;
- gry planszowe i wiele innych.

# Definicja problemu

Trzy elementy potrzebne do zdefiniowania problemu:

1. Baza danych: fakty, stany, możliwości, opis sytuacji.
2. Możliwe operacje: zmiana stanu bazy danych.
3. Strategia kontrolna: start, koniec i kolejność operacji.

Ciąg operacji tworzy sekwencję działań:

od stanu początkowego => do stanu końcowego (celu).

- Z każdą operacją związany jest pewien koszt.
- W procesie szukania należy dążyć do minimalizacji całkowitych kosztów.

# Jakie szukanie?

Znaleziona sekwencja operacji  $\Leftrightarrow$  rozumowaniu.

- **Rozumowanie bezpośrednie:** danych  $\Rightarrow$  celu (data driven), szukanie z dołu do góry (bottom-up search).
- **Rozumowanie wstecz:** celu  $\Rightarrow$  danych, kierowane przez cele (goal directed), z góry na dół (top-down).
- **Analiza środków i celów** (means-ends analysis): strategia mieszana, tworzy się cele pośrednie, szuka „od środka”, dane  $\Leftrightarrow$  cel pośredni  $\Leftrightarrow$  cel.

Jak przedstawić proces szukania?

Grafy lub struktury drzewiaste.

Strategie przeszukiwań:

różne sposoby tworzenia grafów lub wędrowania po grafie.

# Grafy i szukanie

węzły = stany bazy;  
tworzone w miarę potrzeb.

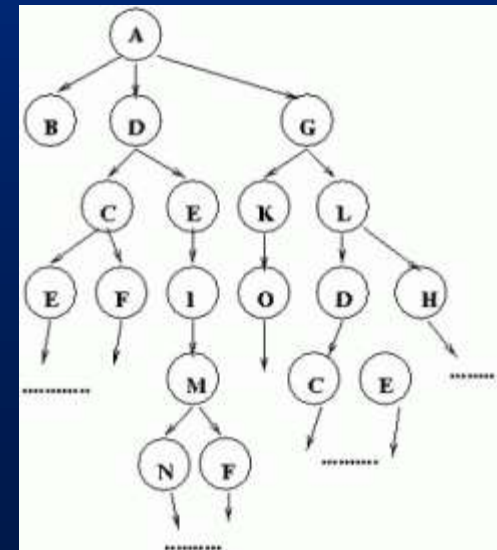
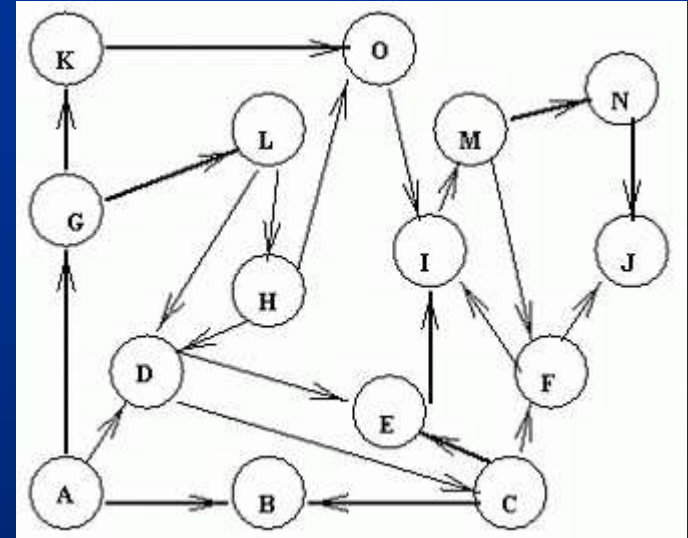
łuki = operacje prowadzące do  
nowych stanów.

Struktury drzewiaste:

grafy w których każdy węzeł ma tylko jednego  
poprzednika.

Drzewo wszystkich możliwości wyznacza  
przestrzeń szukania.

Z grafami wiążą się automaty skończone (FSM,  
finite state machines), dyskretne układy  
dynamiczne oparte na diagramach stanów  
reprezentowanych na grafie.



# Eksplozja kombinatoryczna!

Przestrzeń szukania może być nieskończona lub ogromnie wielka.

10 operatorów, do celu 100 kroków to mamy  $10^{100}$  możliwości.

Dla warcabów jest około  $10^{40}$  różnych gier,  $5 \cdot 10^{20}$  pozycji figur!

Liczba gier w szachach jest rzędu  $10^{120}$  - tyle jest liści drzewa gry! Średnio  $b=35$ , więc 20 ruchów daje  $35^{20} \sim 7.6 \times 10^{30}$  możliwości.

Kostka Rubika  $b=13.3$ , dla 15 ruchów daje ok.  $10^{17}$  możliwości.

Jak znaleźć drogę do rozwiązania w tak wielkiej przestrzeni tworząc najmniejszy graf szukania?

W AI interesują nas zagadnienia nieobliczalne, NP-trudne - liczba węzłów rośnie prowadząc do „**eksplozji kombinatorycznej**”.

Drzewo szukania powinno być małym podzbiorem całej przestrzeni szukania, inaczej szukanie będzie nieskończenie długie. Jak to osiągnąć?

# Heurystyki

Metoda „wygeneruj i testuj”.

Generator nowych stanów (węzłów) produkuje hipotezy.

- generuj wszystkie możliwe stany (zupełność);
- unikaj powtarzania tych samych stanów (unikalność);
- używaj wszystkich informacji pozwalających wstępnie ograniczyć możliwe hipotezy.

Testuj wyniki.

„Heurystyczny” - pomocny w rozwiązaniu, służący odkryciu.

Wiedza heurystyczna - wiedza nie gwarantująca rozwiązania.

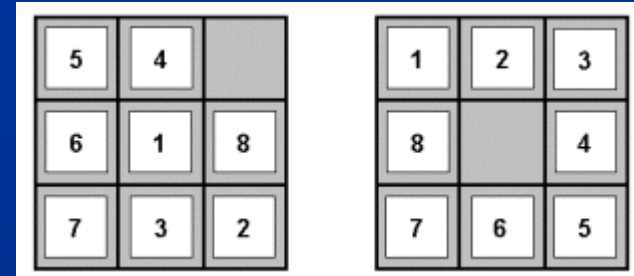
Proces heurystyczny oznacza proces mogący - ale nie gwarantujący - doprowadzić do rozwiązania, strategię, trik, regułę kciuka.

„Heurystyczny” to przeciwstawienie ślepego szukania.





# Przesuwanka



Przykład: 15-ka lub 8-ka, prosta przesuwanka.

Przestrzeń stanów liczy  $9!/2=181440$  elementów

(połowa jest niedostępna bez np. zamiany  $1 \leftrightarrow 2$ ).

Stan = macierz 3 na 3.

Operacje = przesuwanie; 4 operacje na pustym polu;

Ruchy = zbiór operatorów  $O_d, O_g, O_l, O_p$

Zbiór stanów wyjściowych S i końcowych G.

Problem zdefiniowany jest jako trójka (S,O,G).

Rozwiązanie problemu = ciąg operatorów przekształcających S  $\rightarrow$

<https://www.movingai.com/SAS/STP/>

Przykładowe programy sliding puzzle.

Ilustracja kilku algorytmów szukania dla przesuwanki.

# Królowe i skoczki

- Problem N królowych.

Stan początkowy: dowolny układ N królowych.

Operator: przestaw królową na jedno z pustych pól.

Cel: ustawienie N królowych tak, by żadna nie atakowała pozostałych.

Cel dodatkowy: znaleźć wszystkie możliwe rozwiązania.

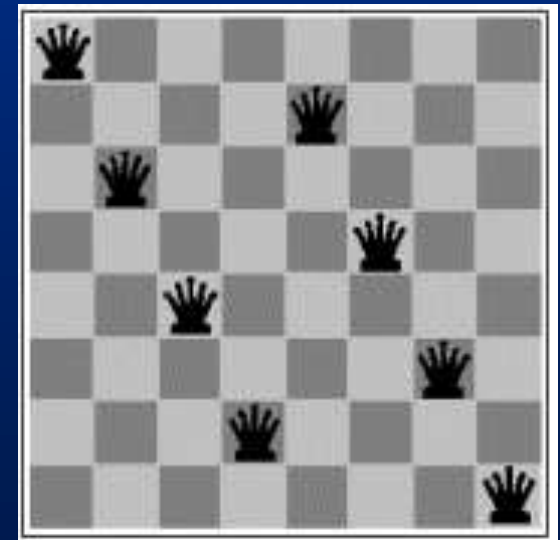
N-Queen Problem – rozwiązania w C.

Inne problemy z użyciem szachownicy:

Problem skoczka szachowego: skacz tak by odwiedzić wszystkie pola ale tylko 1 raz (demo).

Najdłuższa ścieżka skoczka bez przecinania drogi.

Najlepsze znane rozwiązania, wiele wariantów problemów szachowych.



# Lis i gęsi

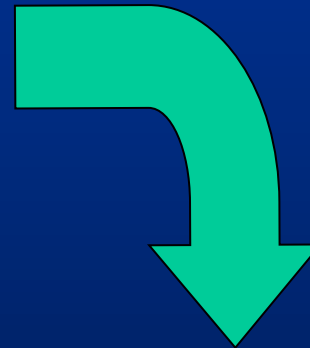
- Jak przewieźć lisa, gęś i ziarno małą łódką na drugą stronę rzeki, jeśli zmieści się w niej nie więcej niż jeden obiekt?

farmer, lis, gęś, ziarno

\_\_\_\_\_

\_\_\_\_\_

pusto



Jeśli nie pilnować to

lis zje gęś,  
gęś zje ziarno  
(farmer zje wszystko).

pusto

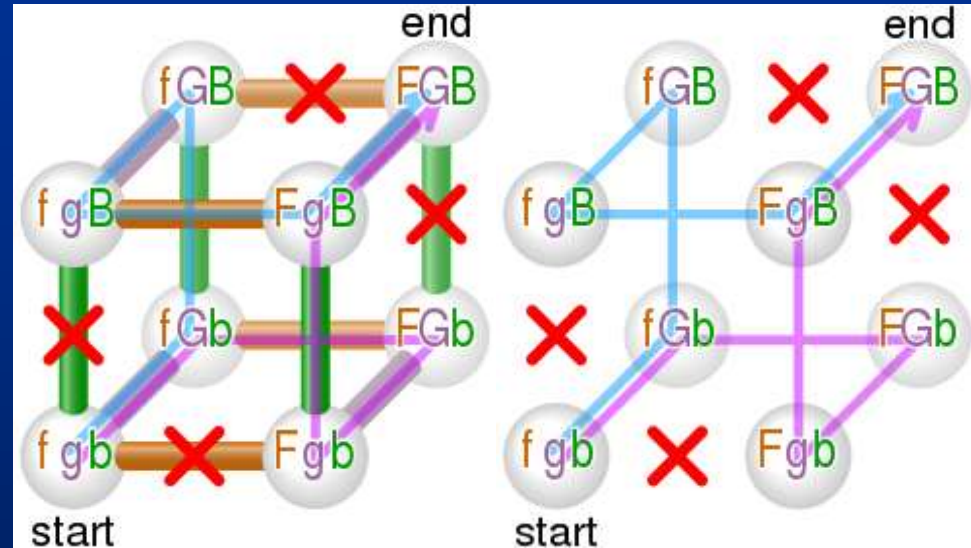
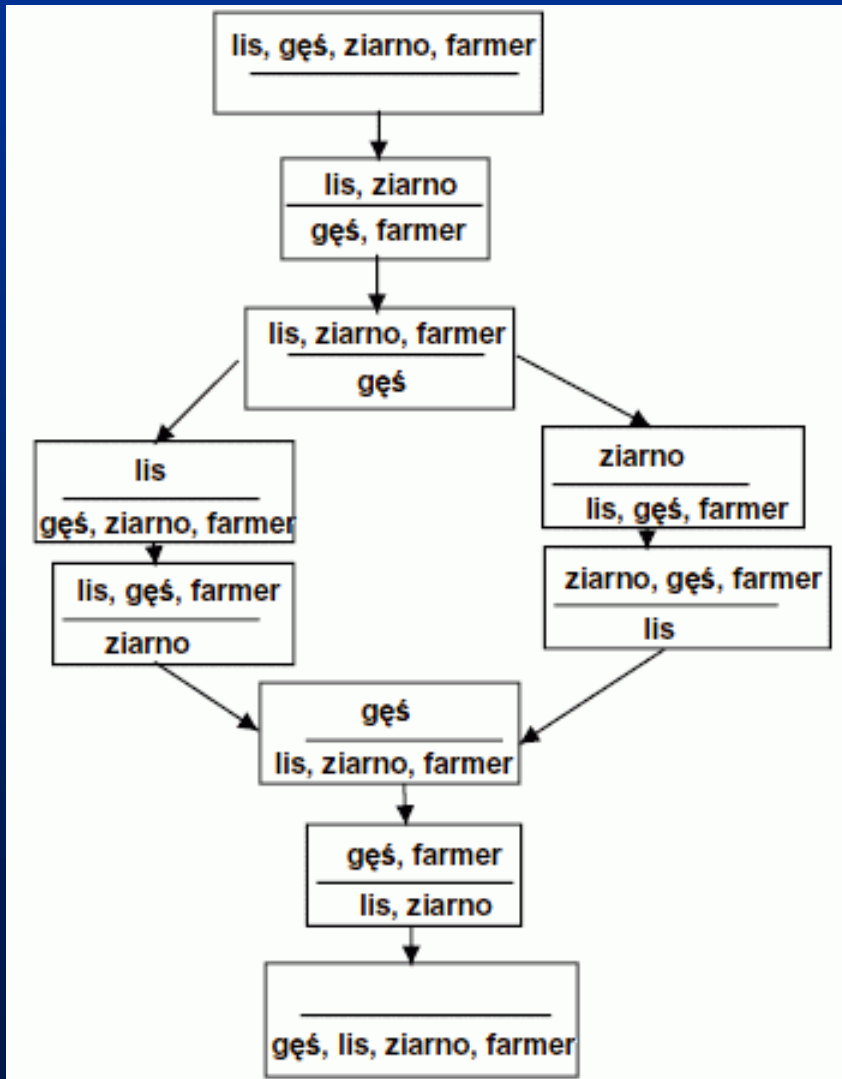
\_\_\_\_\_

\_\_\_\_\_

farmer, lis, gęś, ziarno

# Lis i gęsi

Jak przewieźć lisa, gęś i ziarno małą łódką na drugą stronę rzeki, jeśli zmieści się w niej nie więcej niż jeden obiekt?



Lis i 33 gęsi to popularna gra islandzka, znana jako **halatafl**.

# Kanibale i misjonarze

- Mamy N misjonarzy i N kanibali po jednej stronie rzeki.
- Jeśli na którymś brzegu znajdzie się więcej kanibali niż misjonarzy to zostaną zjedzeni! Łódka mieści tylko 2 osoby.
- Jak ich przewieźć na drugą stronę?  
Spróbuj znaleźć rozwiązanie dla  $N=2, 3, 4$  i  $5$ .



Start:  $[ [m(2),c(2)], [m(0),c(0)], l ]$

Cel :  $[ [m(0),c(0)], [m(2),c(2)], r ]$  Operacje:

- $[ [m(2),c(2)], [m(0),c(0)], l ] \Leftrightarrow [ [m(0),c(2)], [m(2),c(0)], r ]$
- $[ [m(2),c(2)], [m(0),c(0)], l ] \Leftrightarrow [ [m(1),c(1)], [m(1),c(1)], r ]$
- $[ [m(1),c(1)], [m(1),c(1)], l ] \Leftrightarrow [ [m(0),c(0)], [m(2),c(2)], r ]$
- $[ [m(1),c(1)], [m(1),c(1)], l ] \Leftrightarrow [ [m(0),c(1)], [m(2),c(1)], r ]$

Czy możesz stworzyć funkcję, której wartość będzie mała wraz ze zbliżaniem się do celu? Spróbuj najpierw a jeśli się nie uda to obejrzyj na Youtube.

# Zazdrośni mężowie

Bardzo podobny problem do poprzedniego:

Mamy 3 mężów A, B, C i ich żony a, b, c. Chcą przepłynąć na drugą stronę rzeki łódką mieszczącą dwie osoby, ale mąż nie może zostawić swojej żony w towarzystwie obcego mężczyzny.

Mamy takie kroki:

A a B b C c | rzeka | Pusto

B b C c | rzeka | A a

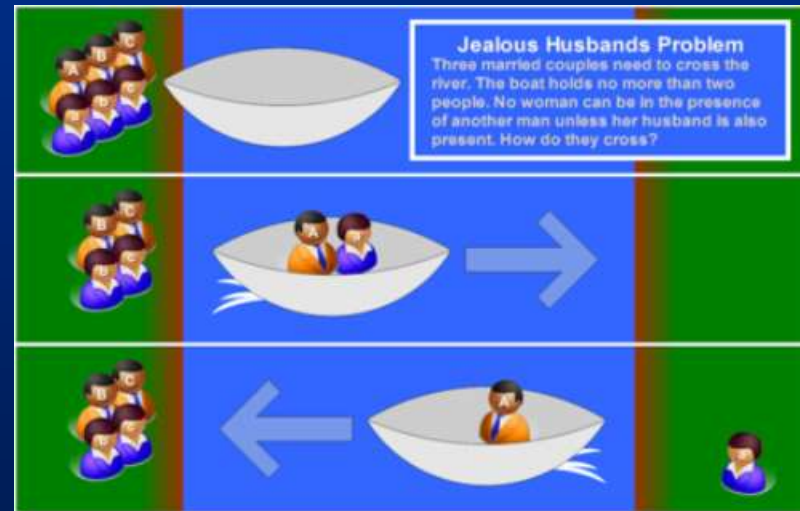
A B b C c | rzeka | a

A B C | rzeka | a b c

A a B C | rzeka | b c

A a | rzeka | B b C c

Co dalej ?



Analiza i przykładowy program

# Kryptoarytmetyka

- Zamień litery na cyfry.

Stan początkowy: słupek arytmetyczny z literami.

Operator: zamień literę na cyfrę, zachowaj jednoznaczność.

Cel: zamień wszystkie litery;  
operacje na cyfrach muszą się zgadzać.

- Przykład:

```
FORTY
+TEN
+TEN
=====
SIXTY
```

- Rozwiązanie:

```
29786
+850
+850
=====
31486
```

Problem znany też jako „verbal arithmetic”, analiza i liczne linki.



# Inne problemy

Szukanie z więzami: 4 kolory mapy.

Czy 4 kolory wystarczą do pokolorowania dowolnej mapy tak by stykające się obszary miały różne kolory? Dla Australii to łatwe.

[https://www.mimuw.edu.pl/~awojna/SID/wyklady/przesz\\_z\\_wiezami.pdf](https://www.mimuw.edu.pl/~awojna/SID/wyklady/przesz_z_wiezami.pdf)

Problem przydziału np. kto którą klasę będzie uczyć?

Problem rozplanowania zadań np. gdzie i kiedy będą odbywać się zajęcia?

Konfiguracja sprzętu, jeśli jest dużo opcji do wyboru.

Problemy logistyki, optymalizacja działania, np.:

- Planowanie budowy autostrady.

- Zaplanowanie etapów produkcji.

Animacja algorytmów szukania.

Tools for Learning Artificial Intelligence – AISpace



# Reprezentacja redukcyjna

Najważniejsze nie są tu stany, ale cele, czyli opisy problemu

- Opis początkowego problemu
- Zbiór operatorów transformujących dany problem na problemy cząstkowe
- Zbiór problemów elementarnych



Wieża z Hanoi.

Krążki A, B, C

Kołki,  $i, j, k$ .

Problem: przesuń  $n$  klocków z  $i$  na  $j$ .      Podproblemy:

- Przesuń stos  $n-1$  klocków z  $i$  na  $j$
- Przesuń jeden klocek z  $i$  na  $k$
- Przesuń stos  $n-1$  klocków z  $j$  na  $k$

Problem elementarny: przesunięcie pojedynczego klocka.

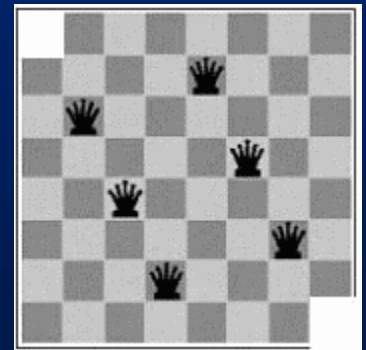
# Wybór reprezentacji

- Szukanie w p-niach problemów lub stanów jest metodą ogólną.
- Jak wybrać odpowiednią przestrzeń?

Odpowiednia reprezentacja to znaczna część rozwiązania:

- uwidacznia istotne relacje;
- ujawnia wszystkie więzów ograniczających możliwe relacje;
- jest zrozumiała, kompletna, zwięzła;
- jest efektywnie wykorzystywana w modelu komputerowym.

Czy 31 domin może pokryć wszystkie pola szachownicy, z której usunięto 2 rogi leżące po przeciwległych stronach?



# Praktyczne problemy

Metody szukania są przydatne do rozwiązywania wielu praktycznych problemów, używamy ich codziennie. Kilka przykładów:

- Szukanie optymalnej drogi: rutowanie pakietów w sieciach komputerowych, rezerwacje lotnicze lub kolejowe.
- Projektowanie VLSI: jak optymalnie rozmieścić miliony elementów uwzględniając wiele ograniczeń?
- Jaką strukturę przyjmie białko po zwinięciu się łańcucha aminokwasów? To pytanie za miliard \$.
- Szukanie drogi przez roboty, szukanie inteligentnego życia na Marsie, autonomiczne urządzenia ratunkowe przeszukujące teren.
- Planowanie zajęć w większej szkole.
- Gry planszowe, gry wojenne, gry komputerowe.
- Dowodzenie twierdzeń matematycznych.
- Problemy wyceny kosztów ubezpieczenia.

# Procedury szukania

- Szukanie na ślepo - nie mamy żadnej informacji.
- Szukanie heurystyczne - potrafimy ocenić postępy.

Na ślepo:

- Monte Carlo, czyli procedura Brytyjskiego Muzeum.  
Dla większych problemów, jeśli istnieje wiele rozwiązań, zwykle można coś znaleźć ... i ślepej kurze ...
- Szukanie w głąb.
- Szukanie wszerek, sprawdzając wszystkie możliwości.

# Literatura

- [Wikibook AI](#)
- L. Bolc, J. Cytowski, Szukanie heurystyczne.
- A. Kisielewicz, Sztuczna inteligencja i logika. Podsumownaie przedsięwzięcia naukowego. Wyd. Nauk-Techniczne 2011
- M. Flasiński, Wstęp do sztucznej inteligencji. WN PWN 2011
- Z. Hippe, Zastosowanie metod sztucznej inteligencji w chemii (PWN, Warszawa 1993)
- J. Mulawka, Sztuczna Inteligencja (1995)
- Jerzy Cytowski, Metody i algorytmy sztucznej inteligencji w cyfrowym przetwarzaniu sygnałów. Akademicka Oficyna Wydawnicza PLJ, Warszawa 1999

# Pytania

- Co to jest przestrzeń szukania? Zdefiniuj, skonstruuj dla zadanego problemu.
- Co to jest heurystyka? Dla problemu X podaj przykładowe heurystyki.
- Jakie są rodzaje szukania i kiedy ich używać?
- Jaki jest największy problem przy szukaniu rozwiązań, podaj przykłady?
- Co to jest reprezentacja w przestrzeni stanu i jak ją skonstruować dla zagadnienia X?
- Co to jest reprezentacja redukcyjna i kiedy ją używamy.
- Jak można uniknąć eksplozji kombinatorycznej i w jakich przypadkach?
- Itp. ...