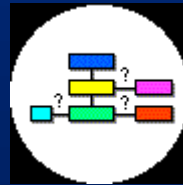


Sztuczna Inteligencja

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: Wlodzislaw Duch

W stronę rosnącej inteligencji

Zrobotyzowana
Automatyzacja
Procesów (RPA)



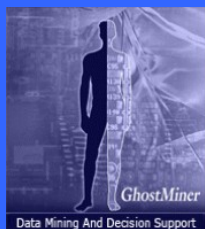
Systemy Ekspertowe



Systemy
kognitywne



Business
Intelligence



LLM, LMM, LAM

GPT Family

LLaMA 1/2 Family

PaLM Family

Gemini, Phi, Mistral, Grok

Agenci, systemy
wieloagentowe

Google Gemini
AI Agents



Potrzebujemy więcej inteligencji, kreatywności, zrozumienia sytuacji.

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.

To problemy dyskretne, jednoznaczne.

Bogaty kontekst wymagany do zrozumienia semantyki słów, obrazów, sterowania przy wielu stopniach swobody, nie da się opisać symbolicznie.

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.

To nie tylko programowalne działania schematyczne, jak w RPA, Robot Process Automation, ale znacznie bardziej elastyczne podejście pozwalające na adaptację do nieprzewidzianych warunków.

Jakie szukanie?

Znaleziona sekwencja operacji \Leftrightarrow rozumowaniu.

- **Rozumowanie bezpośrednio:** 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.

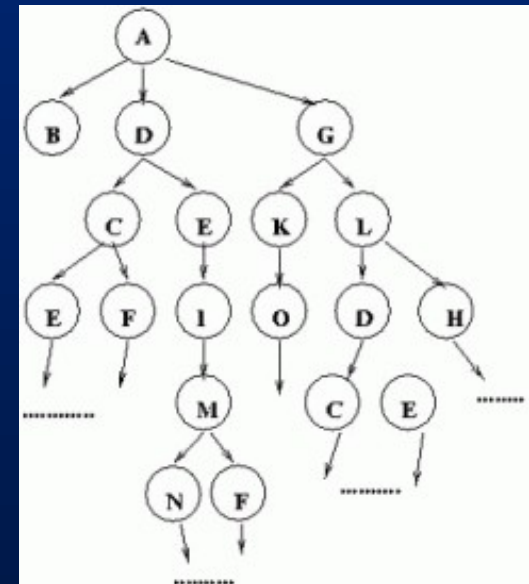
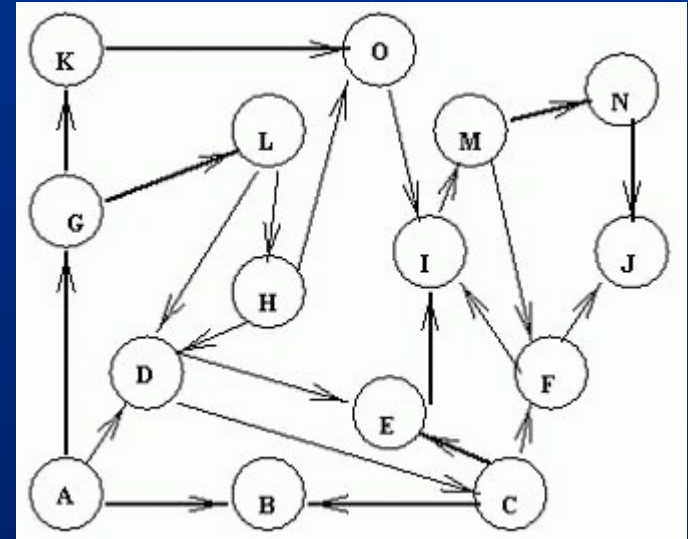
łuki = operacje prowadzące do
nowych stanów.

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.



Ekspluzja 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.

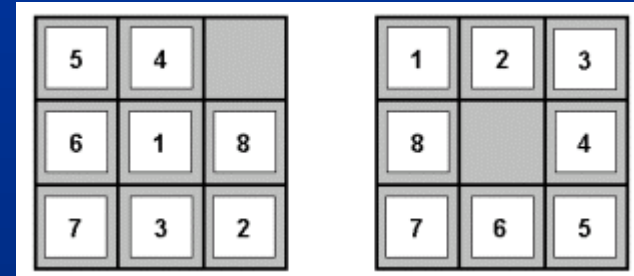
Wiedza heurystyczna - wiedza nie gwarantująca rozwiązania.

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

„Heurystyczny” to przeciwstawienie ślepego szukania.



Przesuwanka



Przykład: prosta przesuwanka 4x4 czyli 15-ka
lub 3x3 czyli 8-ka, z listą publikacji.

Przestrzeń stanów liczy $9!/2=181440$ elementów
(połowa jest niedostępna bez np. zamiany $1 \Leftrightarrow 2$).

Stan = macierz 3 na 3. Dla 4x4 to $16!/2 = 10,461,394,944,000$ stanów.

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 G

A Course on Single-Agent Search: Interactive Demos.

The Sliding Tile Puzzle i kilka innych problemów tego typu.

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.

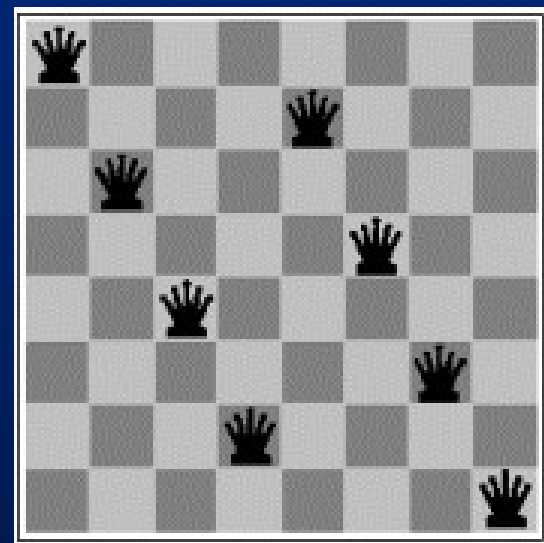
Inne problemy z użyciem szachownicy:

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

Wolfram demonstrations.

Najdłuższa ścieżka skoczka bez przecinania drogi.

Najlepsze znane rozwiązania, wiele wariantów problemów szachowych.

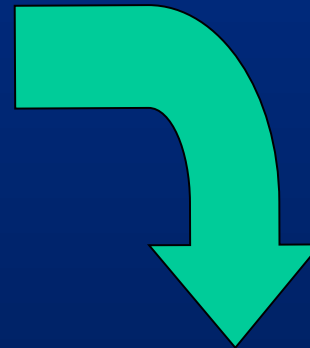


Lis i gęsi

- 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

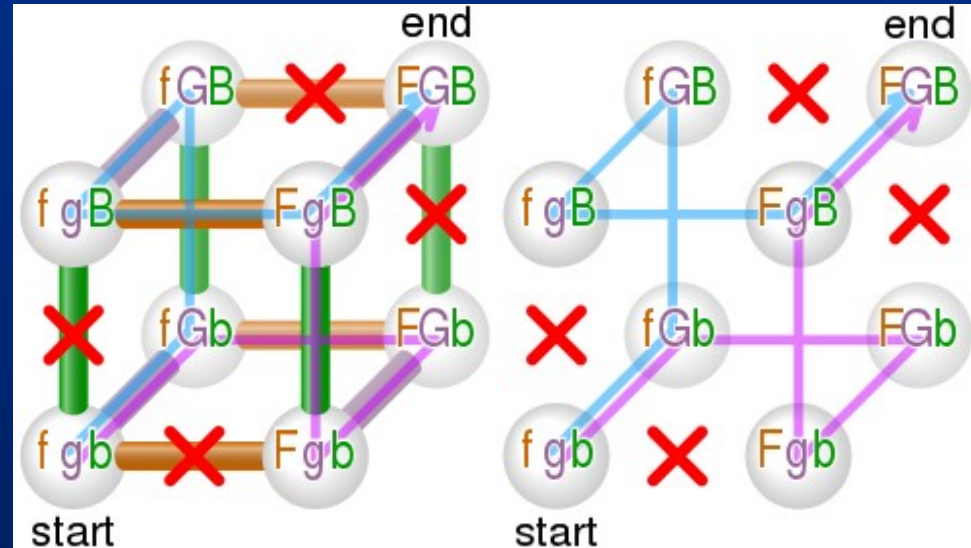
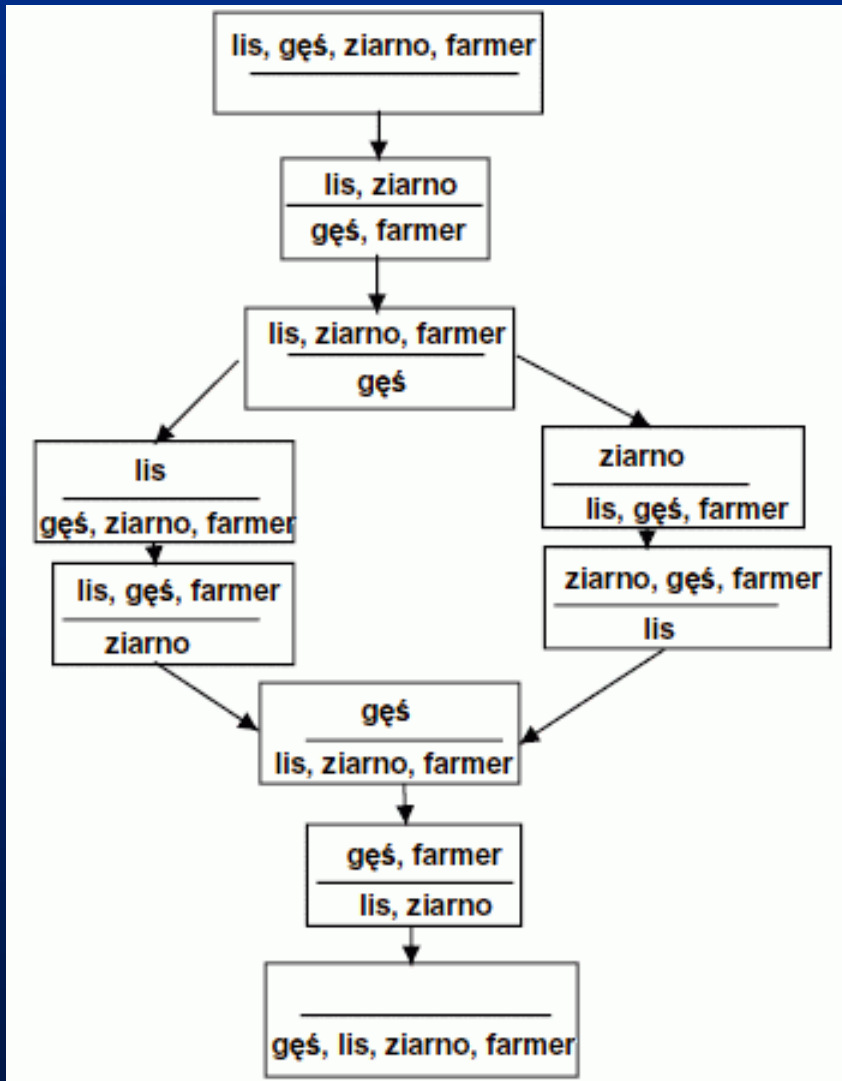
lis zje gęś,
gęś zje ziarno
(farmer zje wszystko).

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)]]

Cel : [[m(0),c(0)] || [m(2),c(2)]] Bezpieczne operacje:

- [[m(2),c(2)] || [m(0),c(0)]] → [[m(0),c(2)] || [m(2),c(0)]]
- [[m(2),c(2)] || [m(0),c(0)]] → [[m(1),c(1)] || [m(1),c(1)]]
- [[m(1),c(1)] || [m(1),c(1)]] → [[m(0),c(0)] || [m(2),c(2)]]
- [[m(1),c(1)] || [m(1),c(1)]] → [[m(0),c(1)] || [m(2),c(1)]]

Czy możesz stworzyć funkcję, której wartość będzie malała wraz ze zbliżaniem się do celu? Spróbuj najpierw dla N=3, 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

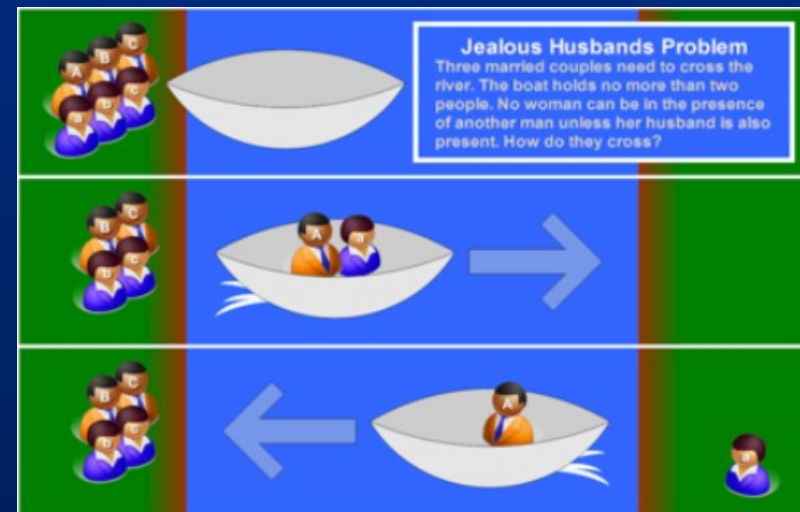
A B b C c | rzeka | a

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

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 kolejne cele, opisy problemu

- Opis początkowego problemu. Klocki na kołku i , a kołki j , k puste.
- Zbiór operatorów transformujących dany problem na problemy cząstkowe, redukcja złożoności.
- Zbiór problemów elementarnych.



Wieża z Hanoi.

Krążki A, B, C

Kołki, i , j , k .

Problem: przesuń n klocków z i na j . Podproblemy, rekursywnie:

- 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.

Przykład wieży z Hanoi



Wieża z Hanoi.

Krążki A, B, C

Kołki, *i, j, k*.

Dla 3 krążków potrzeba 7 ruchów:

1. Przenieś krążek 1 z wieży A na wieżę C.
2. Przenieś krążek 2 z wieży A na wieżę B.
3. Przenieś krążek 1 z wieży C na wieżę B.
4. Przenieś krążek 3 z wieży A na wieżę C.
5. Przenieś krążek 1 z wieży B na wieżę A.
6. Przenieś krążek 2 z wieży B na wieżę C.
7. Przenieś krążek 1 z wieży A na wieżę C.

Udowodnij, że dla n krążków potrzeba $2^n - 1$ ruchów.

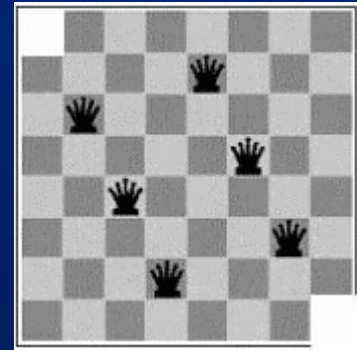
Wybór reprezentacji

- 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?



Nietypowe pytania

Google Interview: zmniejszony do 5 cm wpadłeś do blendera, który włączy się za 60 sekund. Czy się uratujesz?



Myślenie algorytmiczne dla ambitnych, łatwe i trudne zagadki i problemy:

Zbigniew Michalewicz i inn. „Jak to rozwiązać” i „Nauczanie łamigłówek”. E.F. Meyer III i inn.



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.

Zadanie: sprawdzić co ChatGPT robi, a czego nie?

Procedury szukania

Jeśli dobrze sformułujemy zadanie procedura szukania powinna znaleźć rozwiązanie.

- 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 ...

Najprostsze metody:

- Szukanie w głąb.
- Szukanie wszerek, sprawdzając wszystkie możliwości.

Ilustracje algorytmów szukania

- Animacje algorytmów: [VisualAlgo](#)
- Animacje algorytmów szukania na [PathFinding](#)
- Ilustracje [data structures and algorithms](#)
- Visualizations of [data structures and algorithms](#)
- [Algorithm-visualizer](#) z przykładami w Java/C++
- [Data Structures & Algorithms](#) (Google)
- ERIC, [Institute for Education Sciences](#) (publikacje)

Jest wiele animacji takich algorytmów w sieci, może sami znajdziecie ciekawsze.

Literatura

- Wikibook AI

Książki nie są już tak przydatne jak materiały z sieci.

- Mieczysław Muraszewicz (red.), Robert Nowak (red.) Sztuczna inteligencja dla inżynierów. Metody ogólne
- M. Flasiński, Wstęp do sztucznej inteligencji. WN PWN 2011
- A. Kisielewicz, Sztuczna inteligencja i logika. Wyd. Nauk-Techniczne 2011
- Z. Hippe, Zastosowanie metod sztucznej inteligencji w chemii. PWN, Warszawa 1993.
- L. Bolc, J. Cytowski, Metody przeszukiwania heurystycznego. Tom 1,2. PWN, Warszawa, 1989, 1991.
- Jan Antoszkiewicz, Metody heurystyczne. Wyd PWE 1990

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?
- Wyprowadź wzór na złożoność problemu wieży z Hanoi.
- 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.
- Czy można uniknąć eksplozji kombinatorycznej? W jakich przypadkach?
- Czy uczenie na przykładach ma szansę rozwiązać złożony problem kombinatoryczny? Czego brakuje?
- Itp. ...