

■ A mesterséges intelligencia alkalmazási területei Az ötödik generációs számítógép projekt

- **A mesterséges intelligencia alkalmazási területei:**
 - logikai játékok (logical games)
 - tételbizonyítás (theorem proving)
 - automatikus programozás (automated programming)
 - szimbolikus számítás (symbolic algebraic computation)
 - látás, képfeldolgozás (vision)
 - robotika (robotics)
 - beszédfelismerés (voice recognition)
 - természetes nyelvek feldolgozása (natural language processing)
 - korlátozás kielégítés (constraint satisfaction)
 - cselekvési tervek generálása (planning)
 - szakértőrendszerek (expert systems)
 - mesterséges neurális hálózatok (artificial neural nets).
 - adatbányászat (data mining)
 - ágensek, multi-ágensek (agents, multi-agents).

■ Logikai játékok (logical games)

Mottó: "A játék nem játék!"

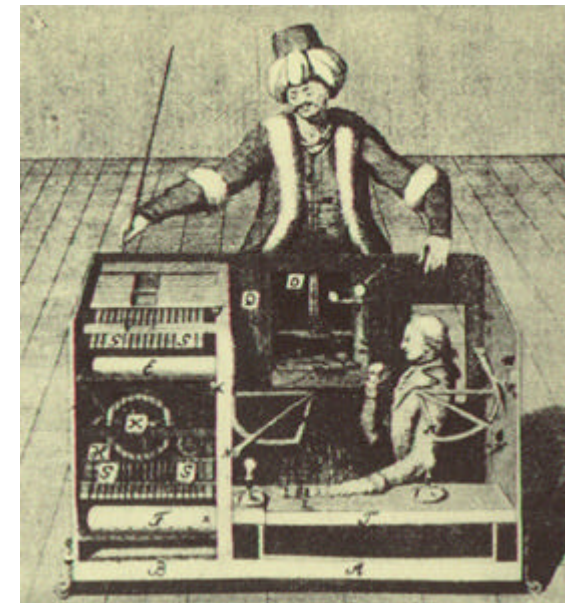


- **A játék lényege:**
 - Ellenérdekű résztvevők felváltva érvényesített stratégiái a végső nyereség érdekében
 - Gazdasági, katonai vetület
 - A véletlennek nincs szerepe
- **Motiváció:** az ember szereti a szellemi kihívásokat (különösen a férfiak szeretnek versenyezni)
- **Fő kérdés:** Lehet-e a gépnek esélye az ember ellen?

■ Játékkal foglalkozó gondolkodók



- **Kempelen Farkas:** sakkautomatája , a Török (csalás) 1769
- **Leonardo Torres y Quevedo:** működő mechanikus sakk-végjátékautomata 1890!!
- **Neumann és Morgenstern:** Theory of Games and Economic Behavior c. könyv
- **Nemes Tihamér:** cikk sakkozó gépről, 1949
- **Claud Shannon:** tanulmány, 1950
- **Alan Turing:** sakkalgoritmus, 1951
- **A játékelmélet alkalmazásának kiteljesedését a számítógép teremtette meg**



<http://www.cowderoy.com/graphics/jpg/tu1.jpg>

■ Kétszemélyes zérusösszegű játékok

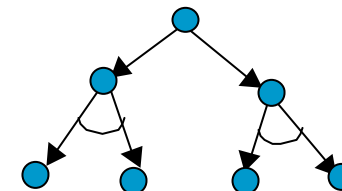
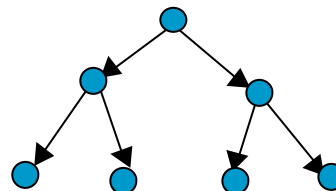
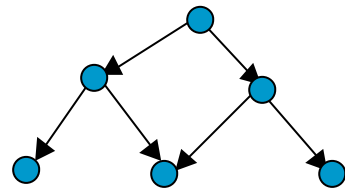


- **Egyik nyer, másik veszít, egyes játékokban döntetlen is.**
- **MAX, MIN**, a két játékos elnevezése
- **Teljes információjú**
 - táblás játékok: igen
 - kártyajátékok: nem
- **Diszkrét:** elméletileg véges számú játékállás, lépés
- **Végteszt:** nyert-e valamelyik? Döntetlen?
- **Nyerő stratégia:** kényszerítve nyer
- **Tökéletes definiáltság és kevés szabály miatt ideális MI tesztterületnek**

■ A játék szemléltetése



- **A játék szemléltetése:** irányított gráffal
- **Gráf-fajták:**
 - **Hálós gráf:** egyféle állás – ugyanazon csomópont
 - **Fagráf:** ugyanazon állásnak több csomópont felel meg (leggyakoribb, jól kezelhető)
 - **ÉS/VAGY gráf** : az egyik játékos szemszögéből. (Saját lépések között : VAGY, ellenfélre között: ÉS.) Csomópontban kell: legalább egy nyerőcímkés VAGY, ill. ÉS esetén mind legyen nyerőcímkés.



■ Gráfjellemzők



- Csomópont \Leftrightarrow játékállás
- Élek \Leftrightarrow legális lépések
- Fagráf-jellemzők:
 - Gyökér csomópont \Leftrightarrow a játék kezdete (elemzés kezdete)
 - Szintek, mélység, depth, d, váltakozva MAX, MIN, gyökér: d=0
 - Levél csomópontok: az egyes játszmák vége, valaki nyer, vagy döntetlen; nyereségérték.
 - Elágazási tényező: a csomópontnak megfelelő állásból tehető legális lépések száma, branching factor, b
 - Átlagos elágazási tényező: több, n csomópontra:

$$\frac{\sum b_i}{n}$$

$$i = 1..n.$$

■ A legjobb stratégia meghatározása



- A legjobb lépés kiválasztása az adott csomópontból megtehetőek közül:

- **Minimax algoritmus:** Neumann János, 1928

- **Stratégiamátrix**

MAX	3	5	1
	9	2	4
MIN			

(MAX a 2. sort választja, mert annak a minimuma nagyobb= 2, nyeregpon)

- Az algoritmusban: MAX a kiinduló éleken visszkapott értékek maximumát, MIN a minimumát rendeli a csomóponthoz.
- Az értékek számítása a fa leveleitől a gyökér irányába.

■ A játék értéke



- **A játék értéke:** A teljes játékfára elvégzett minimax algoritmus eredménye. Egyértelműen megmondja, a kezdő játékos nyeri-e a játékot hibátlan lépésvezetés esetén, és ha igen, azt milyen lépések választásával teheti meg. A nyerés független az ellenérdekű játékos lépéseitől ilyenkor. Ez az algoritmus NEM épít a játékosok szerencséjére, azaz az ellenfél hibázására.
- **A játék teljes fája és a játék értéke csak kis egyszerű játékok (pl. TicTacTo) esetén határozható meg a gyakorlatban.**
- **Gond:** a játék állásainak száma a mélységgel exponenciálisan nő, közelítőleg=
$$b^d$$
- **Kombinatorikus robbanás:** sok a lehetőség, illetve kis teljesítményű a fa kiértékelése.

■ A hatékonyság növelése

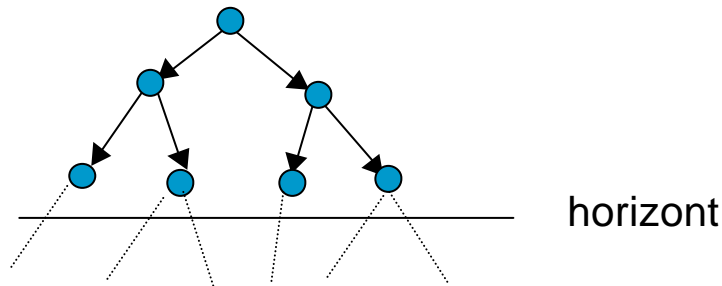


- **Alfa-béta lenyesés:** a fa egyes ágainak levágása. A játék értékére nincs kihatással, de a számításigényt csökkenti. Nem kell kimerítő keresés. Általános esetben kb. 33%-kal mélyebb fa elemezhető ki.
- **Gyakorlatban sokszor még ez is kevés.**
- **Gyakorlati megoldás összetett játékok esetében: az elemzés mélységének egy adott szintre korlátozása.**
- **Gond:** a játéklefolyások teljességükben ismeretlenek, a végzõdés elvésznek, **nincs a minimax algoritmussal felterjeszthetõ érték.**
- **Megoldás** (közelítõ, nem ad feltétlenül azonos eredményt, mint a minimax)(Shannon):
 - **Az elvágással keletkezett leveleknek megfelelõ állásokhoz tapasztalati nyerési esélyértékek rendelése, heurisztikus kiértékelõ függvények (HKF) révén.**

■ A heurisztikus kiértékelő függvény



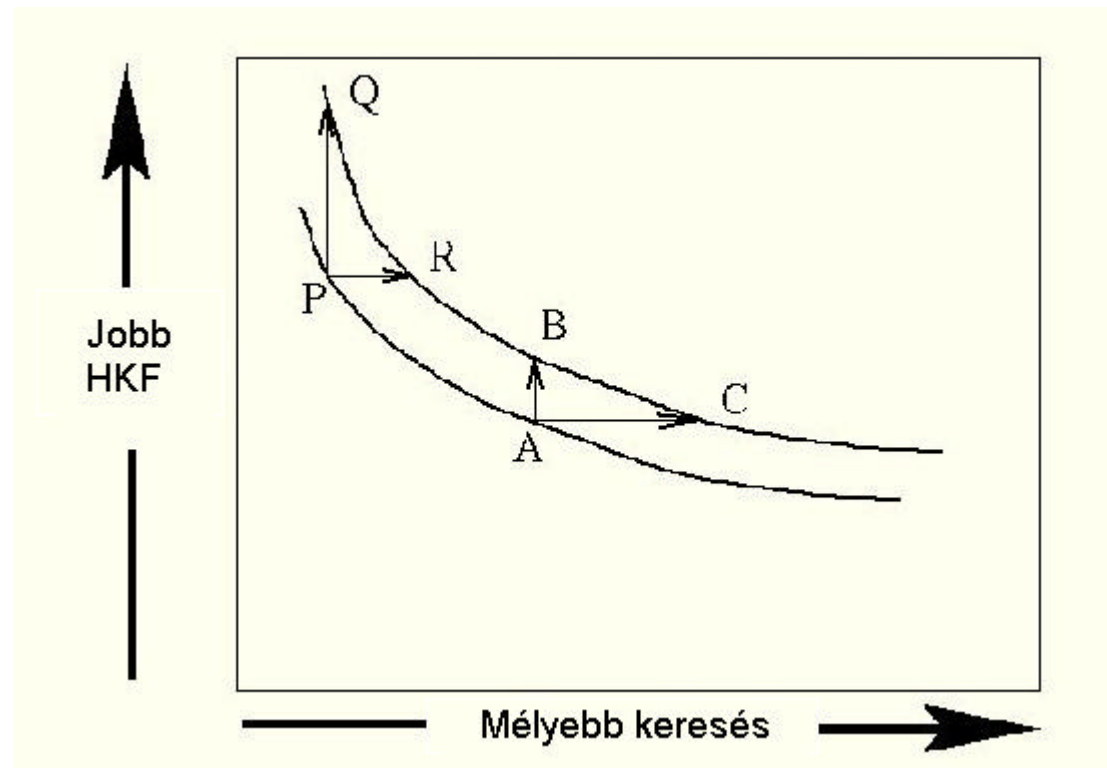
- **Heurisztikus kiértékelő függvény (HKF):** a játékot jól ismerő, nagy tapasztalatú emberek által megadott függvény, mely az állás jellemzőiből számít egy, az állás jóságát megítélő értéket.
- **A minimax algoritmus (és az alfa-béta lenyesés) alkalmazása a HKF értékekre alapozva:** a kapott játékvérték, ill. közbelső érték annyira lesz jó, közelíti a teljes fa esetére adódó pontos értéket, amennyire jó, kifinomult a HKF.
- **Problémák a fa elvágása miatt:**
 - **horizont effektus:** az elvágás alatti részen bekövetkező kedvező, vagy kedvezőtlen hatások nem látszanak (pl. sorozatos sakcadással lemehetünk a horizont alá és nem láthatjuk, hogy a játék a sakkot kapó számára kedvezőbb állású).



■ A heurisztikus kiértékelő függvény ..



- A HKF jóságának és a keresésigénynek az összefüggése



■ További hatékonyságnövelő technikák



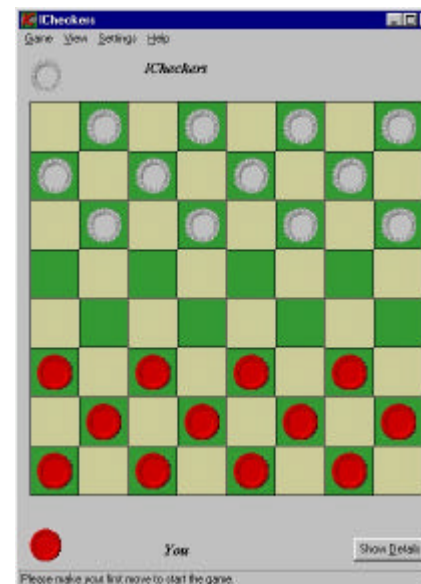
- **B* algoritmus**, intervallumkorlátok a csomóponti értékekre a lenyeséshez
- **SSS* algoritmus**, az alfa-béta lenyeséstől is több ág elhagyása (játékspecifikus)
- **Intervallumtechnika helyett valószínűségi eloszlás**
- **Iteratív mélyítés**: jó az alfa-béta lenyesés rendezettségi igényének részbeni kielégítésére és az időkorlátos játékokhoz is
- **Legalább részleges rendezettség elérése** az alfa-béta lenyesés hatékonyságának növelésére
- **Célhardver és párhuzamos számítás alkalmazása**

■ Híres programok, hardver implementációk

- Legeredményesebbek:
 - Belle, első sakkcélhardver, 1982
 - Chinook, első ember ellen világbajnok játék (dáma)



<http://web.cs.ualberta.ca/~chinook/>



■ Híres programok, hardver implementációk ..



- **Legeredményesebbek:**
 - **DeepBlue, sakk,** (Feng Hsu, Joseph Hoane and Murray Campbell),
1998 Garri Kaszparov ellen nyert. (200millió állás/sec)
 - **Okok:**
 - » **Egycsipes keresőgép**
 - » **Párhuzamos hardver több szinten párhuzamosítva**
 - » **Javított kereső algoritmus**
 - » **Komplex HKF**
 - » **Nagymesterek játszmáit tartalmazó adatbázis.**
 - **Fritz7, sakk:**
a Kaszparovtól a világbajnoki címet
elhódító Vlagyimir Kramnyik kihívója,
2002



■ Híres programok ..



- **Legeredményesebbek:**
 - **Othello:**
 - Logistello, Michael Buro, a legjobban játszó embernél jobb.
 - **Backgammon:**
 - TD-gammon, Gerald Tesauro, világbajnok szintű.
 - **Go:**
 - A legerősebb programok is még csak kissé jobbak a kezdő szintű emberjátékosoktól

■ Tételbizonyítás



- Matematikai tételek bizonyítása az alapaxiómákból kiindulva, pl. kijelentés (propozíciós, nulladrendű predikátum-) kalkulust, illetve elsőrendű predikátumkalkulust használó MI programokkal, a rezolúció módszerével. A rezolúció lényege: lássuk be, hogy a tényekből, szabályállításokból és a bizonyítandó állítás negáltjából álló halmaz kielégíthetetlen, ellentmondásos. Ha sikerül, akkor a bizonyítandó állítás csak igaz lehet.
- A tételbizonyítók **automatikus következtetések** levonására alkalmasak alapaxiómákból, alapinformációkból kiindulva, következtetési szabályokat alkalmazva.
- A tételbizonyítók esetében, szemben a logikai programozási nyelvekkel (pl. Prolog) , **nem csak a logika**, azaz a kiinduló ismeretek, tények és a szabályok **megadása hárul a felhasználóra, hanem a vezérlést**, azaz a következtetés, bizonyítás menetének módszerét **is nyújtania kell**. A tételbizonyítók kutatása elsősorban általánosan alkalmazható vezérlők megtalálását célozza.
- Konkrét alkalmazások: QA1, QA2, QA3 programnyelvek, a QA4 programnyelv procedurális reprezentációt is alkalmaz.

■ Isabelle

- Egy másik professzionális szoftver az **Isabelle**. Tekinthető a következtető rendszerek gyors kreálására alkalmas környezetként, valamint egy komplett tételbizonyítóként.
- Az Isabelle az alábbi módszereket támogatja:
 - elsőrendű logika
 - magasabbrendű logika
 - Zermelo-Fraenkel féle halmazelmélet
 - a Martin-Löf féle típuselmélet egy bővített verziója
 - modális logika
 - a Kiszámítható Függvények logikája.



■ A tételbizonyítók eredményei



- Matematikai tételek, sejtések bebizonyításában
- Szoftver és hardvereszközök helyességellenőrzésében és szintézisében
- Áramkörök tervezésében
- stb.

■ Automatikus programozás

- **Cél:** a szoftverkészítés munkájának automatizálása, olyan eszközök létrehozásával, melyeknél elegendő a megoldandó feladatot specifikálni, a megoldás algoritmus és programja automatikusan készül el.
- Az automatikus programozás esetén a szoftver specifikációja értelemszerűen kisebb és könnyebben megadható, mint maga a program lenne valamilyen programnyelven



■ Automatikus programozás generikus algoritmusokkal

- Az automatikus programozás egyik módszere az általános algoritmusokra (generic algorithms) épít. Az általános algoritmusok egyszerű programozási feladatokra adnak megoldást, például egy sorozatot rendeznek.
- A felhasználó feladata a konkrét programozási feladat és a generikus algoritmusok közötti kapcsolat megadása, pl. hogyan feleltethető meg a generikus algoritmus valamelyik absztrakt adata a konkrét probléma egy jellemzőjének.
- Ha a megfeleltetés adott, egy fordítási folyamat eredményeként egy specifikus algoritmust kapunk, amely már konkrétan megoldja a kitűzött feladatot.
- A generikus algoritmusok és a feladatok egymáshoz rendelése grafikus segídlettel is történhet.

Lásd: <http://www.cs.utexas.edu/users/novak/autop.html>

■ Automatikus programozás evolúciós algoritmusokkal

- Az ADATE (Automatic Design of Algorithms Through Evolution) rendszer nem genetikus algoritmussal állítja elő a feladatot megoldó programokat
- Az algoritmusok nagyméretű kombinációs keresés révén születnek, mely keresés kifinomult program-transzformációkat alkalmaz
- Az ADATE rendszer különösen szimbolikus, funkcionális programok származtatására előnyös
- Windows-os demo program elérhető az alábbi lapról:
http://www-ia.hiof.no/~rolando/adate_intro.html



■ Mintapéllda: lista rendezése

- A feladat-specifikáció, melyet nekünk kell megadni, a következőket tartalmazza:
 1. Az input és az output típusa. Mindkettő céljára egészek listája választható.
 2. Az előredefiniált konstansok és függvények, azaz a primitívek. Ezek vannak kiválasztva az üres listára, egy függvény, amely előre szűr be egy elemet a listába és a < függvény egészekre értelmezett alakja.
 3. Input-output párok. Ezek többféle különböző módon választhatók. Pl:
 - ([], [])
 - ([0], [0])
 - ([0, 1], [0, 1])
 - ([1, 0], [0, 1])
 - ([0, 1, 2], [0, 1, 2])
 - ([0, 2, 1], [0, 1, 2])
 - ([1, 0, 2], [0, 1, 2])
 - ([1, 2, 0], [0, 1, 2])
 - ([2, 0, 1], [0, 1, 2])
 - ([2, 1, 0], [0, 1, 2])

■ Mintapélda: lista rendezése ..

- Az eredmény:

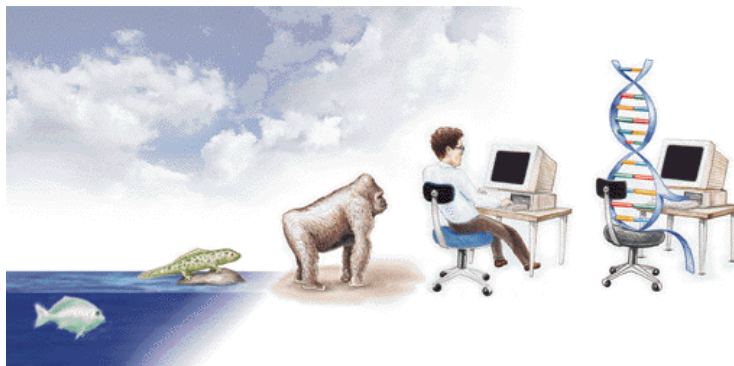
Az **ADATE** számtalan, a specifikációt kielégítő programot generál. A legrövidebb, mely feltehetően a legáltalánosabb is, az alábbi:

```
fun f (V1_0) =  
  case V1_0 of  
    nil => V1_0  
  | (V60_0 :: V61_0) =>  
    let  
      fun g471891_0 (V471892_0) =  
        case V471892_0 of  
          nil => (V60_0 :: nil)  
        | (V471883_0 :: V471884_0) =>  
          case (V60_0 < V471883_0) of  
            true => (V60_0 :: V471892_0)  
          | false =>  
            (V471883_0 :: g471891_0( V471884_0 ))  
    in  
      g471891_0( f( V61_0 ) )  
    end
```

- Ez a program a Standard ML programnyelven íródott.
- A g471891_0 rekurzív segédfüggvényt az ADATE rendszer találta fel. Ez a függvény egy elemet szűr be a rendezett listába.

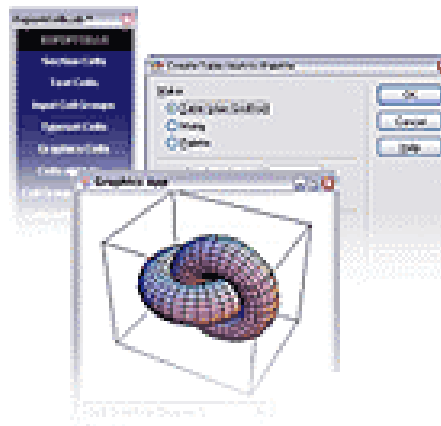
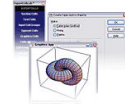
■ Automatikus programelőállítás genetikus algoritmussal

- A genetikus programozás a probléma egy magasszintű meghatározásából kiindulva képes automatikusan előállítani működő számítógépi programokat.
- Véletlenszerűen generált ezernyi ősprogram halmazából indulva, a programok populációja folyamatosan javulva fejlődik sok generáción át. Az **evolúciós keresés** a legrátermettebb és természetesen felbukkanó mintázatokkal rendelkező műveletek túlélésének darwini elméletét alkalmazza, köztük a keresztezést (szexuális rekombinációt), a mutációt, génduplikációt, géntörlést, valamint bizonyos fajtáit a fejlődési folyamatnak amelyek által az embriók kifejlett organizmusokká váltak.

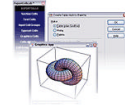


■ Szimbolikus számítás

- Matematikai levezetések, algebrai manipulációk, deriválás, integrálás azonosságainak, trigonometrikus, logaritmikus, stb. azonosságoknak az alkalmazása szimbolikus alakban adott feladatok megoldására.
- Ismertebb szimbolikus algebrai szoftverek: MACSYMA, REDUCE, CAMAL, LAM, ALTRAN, FORMAC, SYMBOL, MATHEMATICA



■ A MATHEMATICA szimbolikus számítási program



- Egyaránt végez numerikus és szimbolikus számításokat

In[1]:= 3 + 62 - 1

In[2]:= 3x - x + 2

Out[1]= 64

Out[2]= 2 + 2x

- Automatikusan végzi az alapvető egyszerűsítéseket:

In[4]:= x² + x - 4 x²

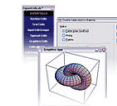
Out[4]= x - 3 x²

- Összetett kifejezésekre addig alkalmazza az átalakítási szabályokat, amíg a kifejezés tovább nem változik, egyszerűsödik

■ A MATHEMATICA további jellemzői

- A problémák definiálására fordíthatjuk az időt, nem a levezetések és számítások elvégzésére
- A világ legnagyobb matematikai eszközkészletét, tudását nyújtja
- Az eredményeket grafikusan is szemlélteti
- Numerikusan pontos
- Szimbolikusan kezeli a feladatokat
- Automatikusan megválasztja a legjobb megoldási módszert a megadott feladathoz
- Jegyzőkönyvbe szervezi a megoldás útját a megadástól a megoldásig
- Nyomdai minőségben prezentálja a kimeneteket
- Többféle formátumban kommunikálhatunk vele, akár a web-en keresztül is
- Gyors alkalmazásfejlesztést tesz lehetővé

<http://www.wolfram.com/products/mathematica/newin42/>



Mathematica programs allow a unique combination of mathematical and computational notation

These definitions correspond to a recently discovered approximation to the number of primes.

```
Ω[n_] := Apply[Plus, Map[Last, FactorInteger[n]]]
μ[n_] := MoebiusMu[n]
```

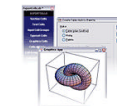
$$P[x] := - \sum_{k=1}^{\lfloor \log(2, x) \rfloor} \mu[k] \sum_{n=2}^{\lfloor x^{1/k} \rfloor} \mu[n] \Omega[n] \left\lfloor \frac{x^{1/k}}{n} \right\rfloor \quad ; x > 0$$

This compares the approximation with the built-in PrimePi function.

```
In[5]:= {P[1000], PrimePi[1000]}
```

```
Out[5]:= {168, 168}
```

■ A MATHEMATICA további jellemzői ..



- Hatalmas mennyiségű integrálból, differenciálegyenletből, összegzésekből és transzformációkból álló kifejezések kezelésére képes, mely ember számára megterhelő lenne, nem beszélve a hibalehetőségről.

Mathematica can do vastly more integrals, differential equations, sums, and transformation than were ever before possible for either humans or computers

$$\int (x^2 - 1) \tan^{-1}(x^2) dx$$

$$\frac{1}{3} (x^2 - 3) \tan^{-1}(x^2) x - \frac{2x}{3} - \frac{\log(x^2 + \sqrt{2}x + 1)}{3\sqrt{2}} +$$

$$\frac{1}{3} \sqrt{2} \tan^{-1}\left(\frac{2x - \sqrt{2}}{\sqrt{2}}\right) + \frac{1}{3} \sqrt{2} \tan^{-1}\left(\frac{2x + \sqrt{2}}{\sqrt{2}}\right) +$$

$$\frac{\log(-x^2 + \sqrt{2}x - 1)}{3\sqrt{2}}$$

$$\int \frac{1}{\sqrt{x^6 + 1}} dx$$

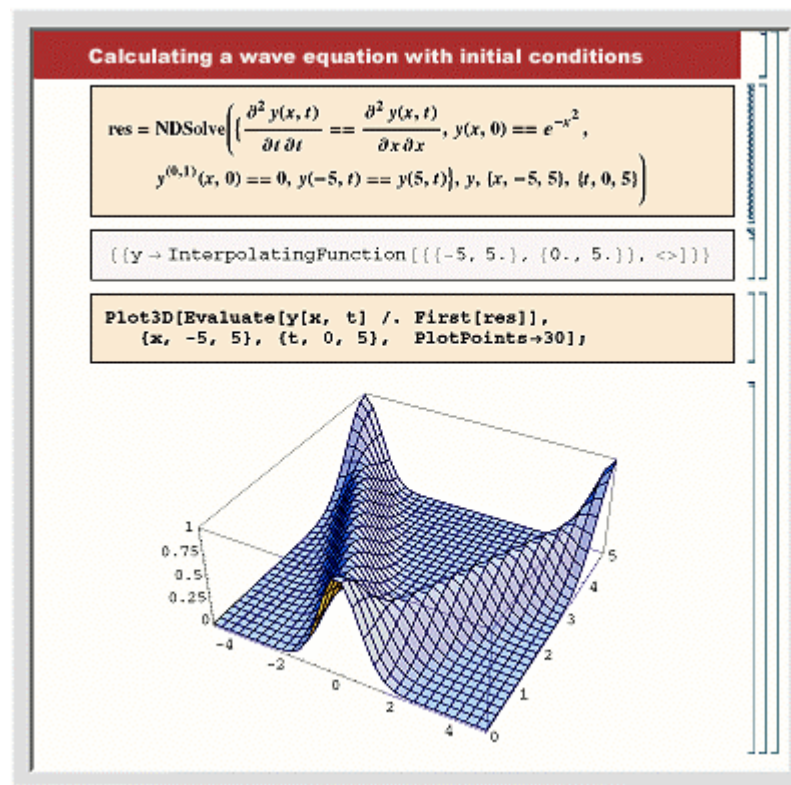
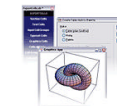
$$\frac{1}{2\sqrt[3]{3} x \sqrt{x^6 + 1}} \left(\sqrt{\frac{x^2(x^2 + 1)}{((1 + \sqrt{3})x^2 + 1)^2}} \left((1 + \sqrt{3})x^2 + 1 \right)^2 \right.$$

$$\left. \sqrt{\frac{x^4 - x^2 + 1}{((1 + \sqrt{3})x^2 + 1)^2}} F\left(\cos^{-1}\left(\frac{(1 - \sqrt{3})x^2 + 1}{(1 + \sqrt{3})x^2 + 1}\right) \middle| \frac{1}{4}(2 + \sqrt{3})\right) \right)$$

<http://www.wolfram.com/products/mathematica/newin42/>

■ A MATHEMATICA további jellemzői ..

- Parciális differenciálegyenletek megoldására, az eredmények szemléltetésére képes.



<http://www.wolfram.com/products/mathematica/newin42/>

■ Gépi látás, képfeldolgozás

- Az ember a külvilágból szerzett információknak a legnagyobb részét látással szerzi meg. Ezért fontos az MI számára a mesterséges látás.
- **A feladat:** Adott egy kétdimenziós bit-térkép, ebből kiindulva meg kell adni a kép leírását, beleértve az alakzatok, méretek, színek, helyzetek paramétereit. Lényegében a nagyon alacsony szintű vizuális adatból egy magas szintű absztrakciót kell elérni, mely megfelel a képen látható objektumoknak.
- Fontos mozzanatok: az emberi látás jellegzetességeinek modellezése, pl.: élek, kontúrok detektálása. A legjobb eredményekkel a neurális hálók kecsegtetnek (pl. PERCEPTRON, *Rosenblatt* kísérlete; karakterfelismerő programok).
- Az elért eredmények magukba foglalják az elektronikus renehártyát is, mely a renehártya sok funkcióját, köztük a látvány elsődleges feldolgozását is modellezi. Létrehozása megkönnyíti a mesterséges neurális hálók bemenőjeleinek előállítását.



1.

1. www.cnn.com/TECH/9604/10/t_t/darts_eyes_lg.jpg

■ Gépi látás, képfeldolgozás ..



- **Input:** kétdimenziós bitmap fájl alakjában a látvány, a kép.
- **Output:** a felismert objektumok és térbeli viszonyuk, fizikai jellemzőik.
- **A képfeldolgozás lépései:**
 - Élek detektálása
 - Mélység meghatározása
 - Alak meghatározása az árnyékoltságból
 - Vonalak címkézése
 - Objektum beazonosítás, helyzet meghatározás



www.toytent.com/1140.html

■ Gépi látás, képfeldolgozás ..



Élek detektálása:

- Az élek fontossága: objektumhatárok, körvonalak, változások a fényvisszaverésben, világításban, mélységben és felületorientációban
- A zaj problémája: ál-élek, melyeket előbb el kell simítani
- Simítás egy pont-szórás függvényen (konvolúció)
- Él megtalálása a fényességváltozás változásának (második derivált) nullát metsző helyeit keresve
- A simítás és az éldetektálás kombinálása végrehajtva egy pont-szórás függvényen való módosítás révén, amely átlagol és megtalálja a fényességváltozások változásait
- Az éldarabkák megcímkézése orientáció szerint egy együttműködési algoritmussal
 - Elemek halmazait és szomszédos elemek közötti kapcsolatokat értelmezzük
 - Az algoritmus ismétlése változtatja az értelmezését minden egyes elemnek, hogy a legnagyobb harmóniában legyen a szomszédos elemekkel (korlátozás kielégítés/kiterjesztés)
 - Párhuzamos feldolgozás: egy processzor elemenként: a processzorok csak a szomszédokkal kommunikálnak.

■ Gépi látás, képfeldolgozás ..



Mélység meghatározása:

- Sztereo egyenlőtlenség: igényli az egymásnak megfelelő pontok megtalálását a képeken
- Objektumok mozgása
- Szem mozgása

Alak meghatározása az árnyékoltságból

- Árnyékoltság
- Ismeretek a fényelnyelésről
- Ismeretek a relatív méretekről

■ Gépi látás, képfeldolgozás ..



Vonalak címkézése :

- Valós élek problémája: konvex, konkáv, kontur: két “irány”
- Pszeudo élek: jelek, árnyék “élek”, -törések
- Kapcsolódás típusok: három sík lehetséges csúcsalkotásai
- Érkezés egy lehetséges címkézett kapcsolódáshoz a következő alakzatok egyikéből észlelhető: egy, három, öt, vagy hét kockából tevődnek össze, melyek különböző irányokból érkeznek
- Lehetséges kapcsolódásfajták: 18
- A címkézést a határoknál kezdjük, felhasználva a megszorításokat melyek a nyíl és villa alakú kapcsolódásokra vonatkoznak.

■ Gépi látás, képfeldolgozás ..



Objektum meghatározás Mintaegyezés vizsgálattal :

- **A régi módszer:** néző-központútól az objektum-központú felé haladó reprezentációkkal
- Éldetektálás, élkapcsolás.
- Alakmeghatározás az árnyékoltságból, vonalcímkezés
- A kiterjedés leírást az ismert objektumok könyvtárában található leírásokkal összevetik
- **Az új módszer:** az objektumleírás kerülése; az alak impliciten benne van az objektum nézeteinek halmazában
- Minden egyes ismert objektumra a könyvtár tartalmaz egy **modell**t, képek halmazát, melyek mindegyike beazonosítható **jellemző pontokat** tartalmaz
- Minden egyes ismeretlen objektumhoz minden egyes modellt párosítva megvizsgálják a jellemző pontok illeszkedését
- Az illeszkedéseken alapulva, jóslásokat végeznek arra nézve hogy további pontoknak hol kellene megjelenniük az ismeretlen objektumon, elkészítve így egy **testreszabott mintát** minden egyes modellhez
- Az ismeretlen objektumot úgy azonosítják, mint egy egyedét annak a modell mintához tartozó kategóriának, amelynek a testreszabott mintájához a legjobb egyezést mutatta az objektum.

■ Gépi látás, képfeldolgozás ..



Felfogás (Felismerés, Megismerés, Megértés)

- Szegmentálás
Hogyan tagolható a látvány régiókra vagy objektumokra?
- Aggregáció
Hogyan kapcsolhatók egybe a részek?
- Bizonytalanság
Hogyan értelmezhetők a bizonytalan részei a látványnak?
- Változatlanság és kategóriák
Minden egyes absztrakt kategóriának számtalan megjelenési formája lehet a látványban. Hogyan lehet az egyformákat, melyek a kategóriát jellemzik, elkülöníteni?

■ Robotika

- Ipari robotok érzékelő rendszereinek, beavatkozó szerveinek, tanítási lehetőségeinek, adaptív képességeinek fejlesztése tartozik ide, szoros kapcsolatban a mesterséges látással.
- A robot témakör azért fontos a mesterséges intelligencia számára, mert a robot rendelkezik az ágenstől is elvárt érzékelés, beavatkozás képességekkel a szenzorai, manipulátor-karjai, kezei révén.
- A robotvezérlő intelligenciájának növelésével nemcsak emberhez hasonlóan gondolkodó, Turing tesztet teljesítő gépi intelligencia jöhet létre, hanem az **emberhez hasonlóan cselekvő, humanoid, emberszerű gépi intelligencia** is.
- Az ily módon létrejövő gépi intelligencia hasznosíthatósága (és veszélyessége) is megnő.

Boilerplate, mechanikus robot, 1879

<http://www.bigredhair.com/boilerplate/soldier/index.html>



■ Sprawlita

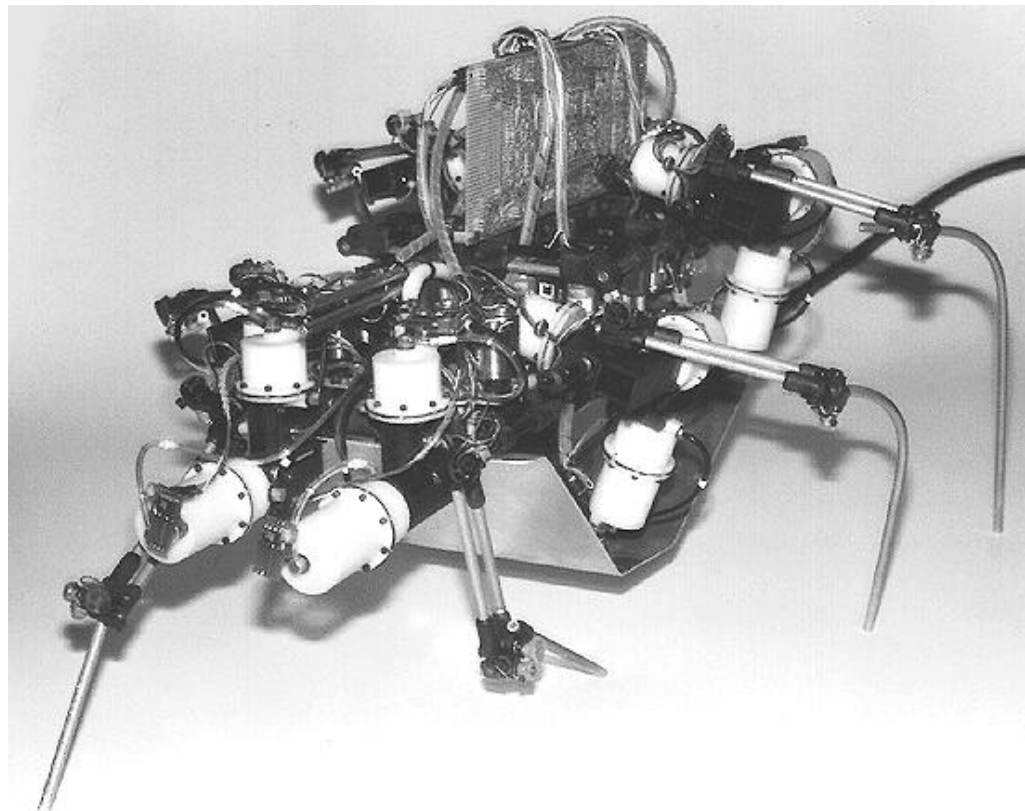
- Lépegető bogár-robot 6 lábbal
- Három testhossz/másodperc sebességgel mászik, akár akadályokon át is.



<http://www-cdr.stanford.edu/biomimetics/documents/sprawlita/sprawlita.html>

■ Boadicea

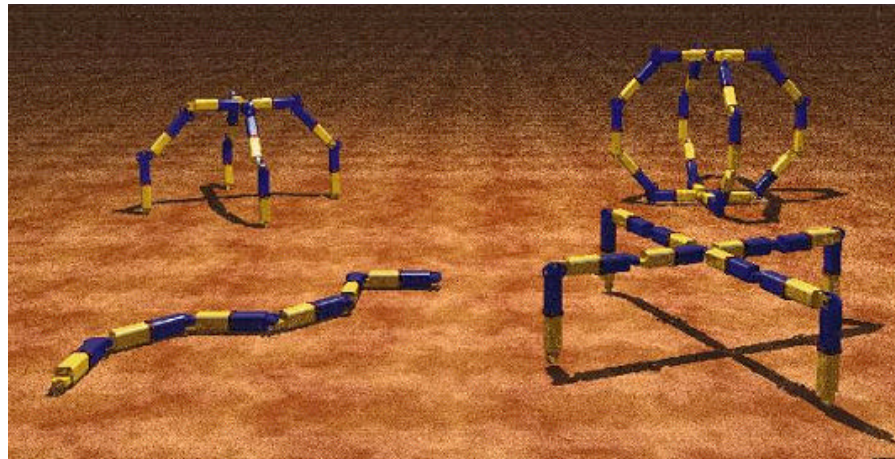
- Kisméretű pneumatikus mászó robot
- Hat lába egyenként három szabadságfokú.



<http://www.ai.mit.edu/projects/boadicea/boadicea.html>

■ CONRO

- Miniatűr újrakonfigurálható robot
- Azonos modulokból áll, amelyeket arra lehet programozni, hogy megváltoztassák a topológiájukat a környezetben adódó olyan kihívásokhoz, mint például egy akadály.
- Az alaptopológia egyszerű kígyó alak, de a rendszer képes újrakonfigurálni magát hogy lábakat növelessen, vagy egyéb speciális nyúlványokat. Minden egyes modul tartalmaz egy CPU-t, memóriát, elemet, micro-motort és változatos szenzorokat és képességeket, köztük látást, huzalnélküli kapcsolatot és a dokkolás szenzorait.



<http://www.isi.edu/conro/>

■ GRACE: A Társasági Robot

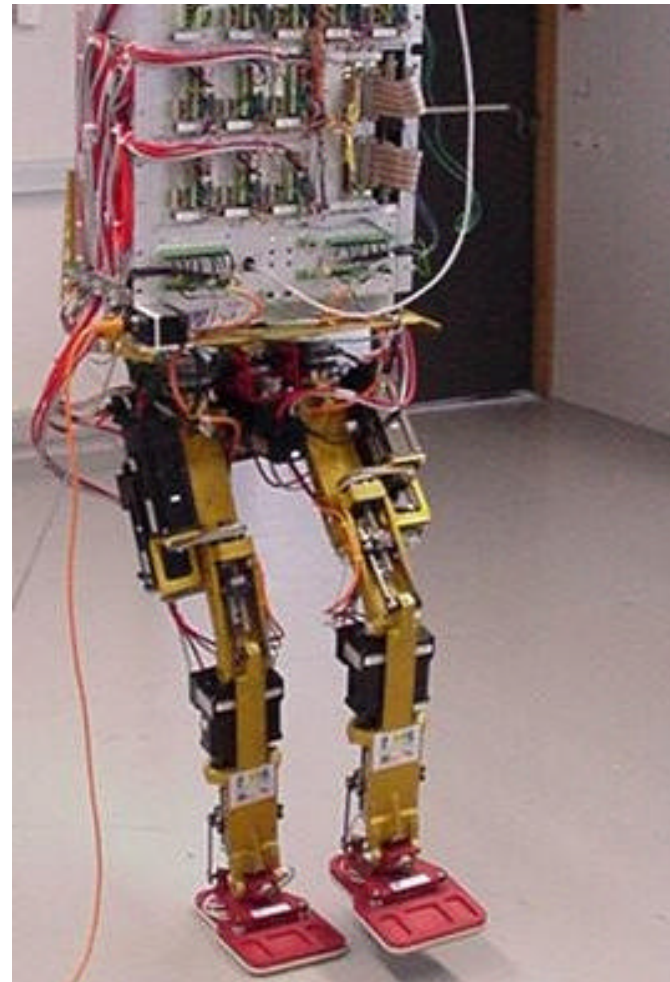
- GRACE egy mobil robot, kifejező arccal és hatalmas mennyiségű szenzorral. A szenzorok között találunk mikrofont, tapintó szenzorokat, infravörös érzékelőket, hangérzékelőt, letapogató lézert, sztereo kamerapárt és egy színes zoom-olható kamerát. Tud beszélni egy kiváló beszédsszintetizátorral és megérti a válaszokat a mikrofonja és beszédfelismerő szoftvere révén.
- GRACE elment egy konferenciára, sorban állt (egy előtte álló hölgyet kitúrta), regisztráltatta magát, majd megtartotta a magáról szóló előadást.



<http://www.palantir.swarthmore.edu/GRACE/>

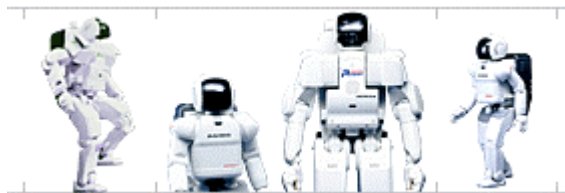
■ BIP a lépegető robot

- A 15 szabadságfokú robot két lábon jár.

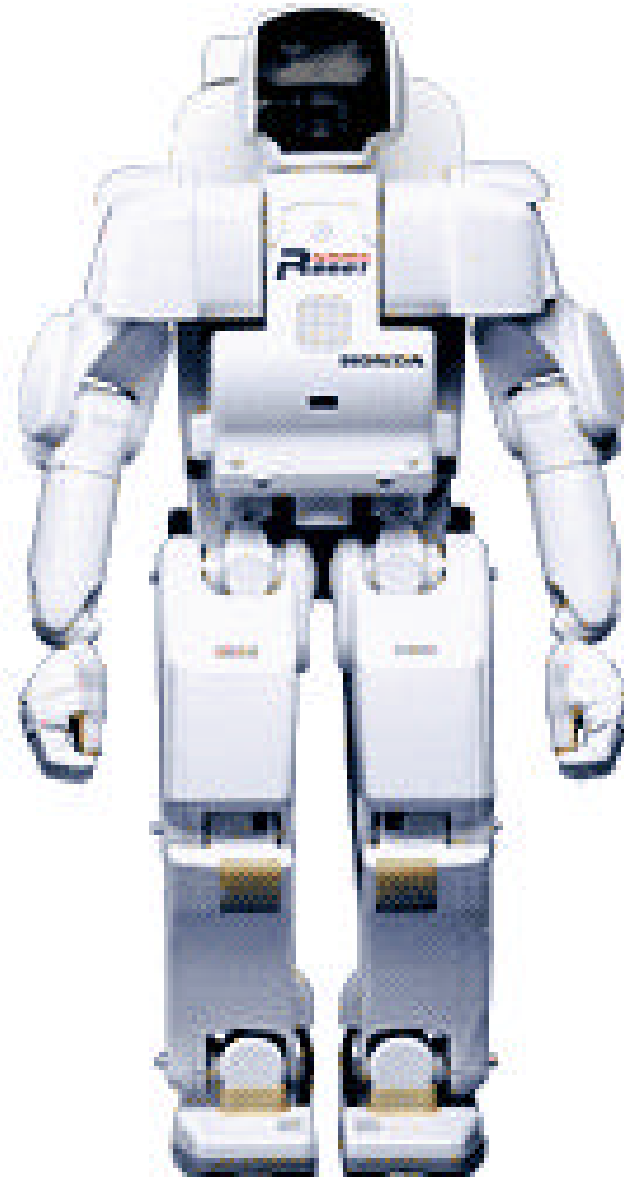


■ A HONDA humanoid robotja

- A honda robot igen fejlett mozgásképeséssel bír.
- Indul a humanoid robotok labdarúgó világbajnokságában is.



<http://world.honda.com/robot/>



■ Autonóm jármű

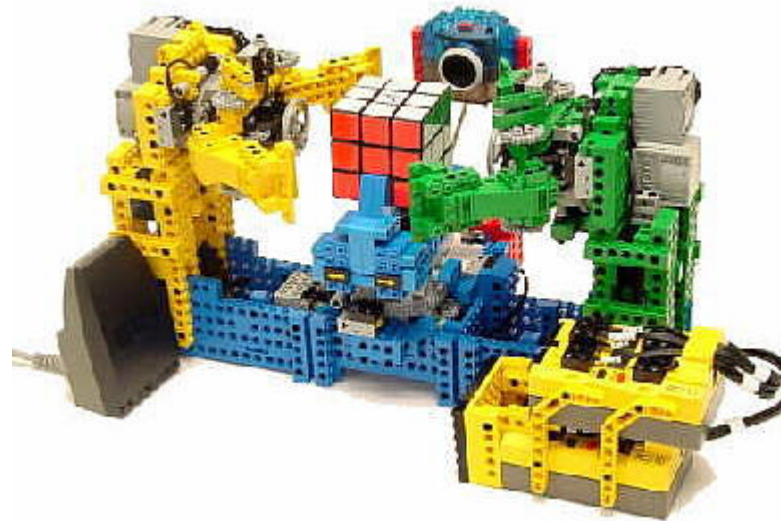
- **Jellemzők:**
- A vezető ébrenlétének figyelése
- Automatikus útkövetés
- Automatikus akadályérzékelés és kikerülés
- Autonóm vezérlés egy úttalan vidéken.



<http://www.syseng.anu.edu.au/rsl/car/>

■ Rubik kocka kitekerő robot

- **Jellemzők:**
- Lego robot alkatrészekből készült.
- Maximum 40 lépésből megoldja a feladatot.



<http://jpbrown.i8.com/cubesolver.html>

■ Beszédfelismerés



- **Cél:** Az emberi beszéd gép által kezelhető, szöveges formára alakítása, végső célként, a nyelvfeldolgozással egyesítve a beszélt nyelv gépi megértése céljából.
- Kezdeti eredmények: szűk szókincs, vagy ugyanazon beszélő esetén nagyobb szókincs felismerése.
- Az emberi nyelvek kb. 50 fonémát, beszédhangot különböztetnek meg.
- Szegmentálás, szavakra bontás → gond: a beszéd szavai egybefolynak.
- A fonémák és a szavak leírására használt betűk között nincs teljesen egyértelmű megfeleltetés → gond. (Pl: eltérő fonémák, azonos betűk: Mondd, Zsanett, kell findzsa? Vagy: azonos fonémák, eltérő betűk: Mi hájjal kenjük a kenyeret, Mihály.)

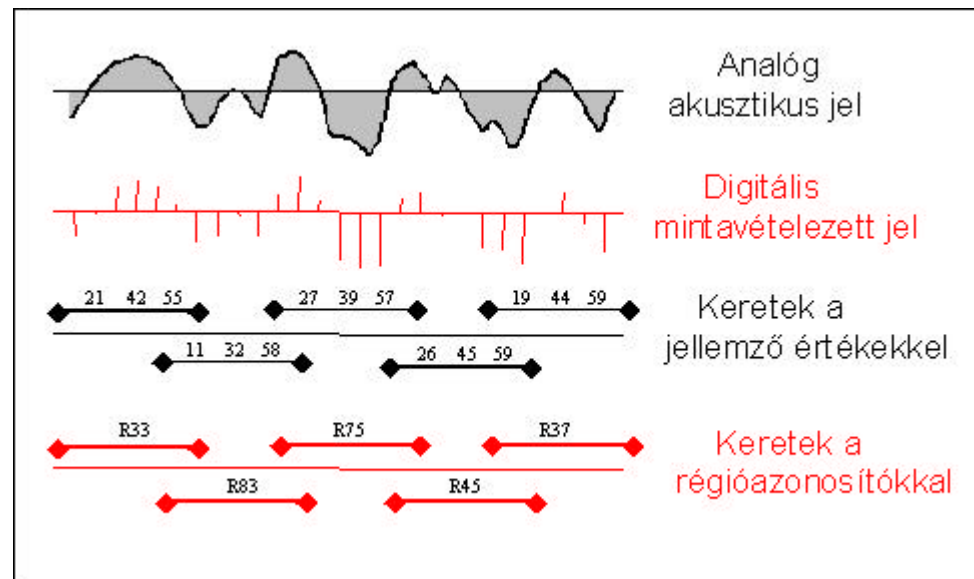
■ A hangjel feldolgozása



Cél: az információ csökkentése és a jellemzők kiemelése.

Lépések:

- Mintavételezés, kvantálás
- Jellemzők kinyerése, keretekben, azonos időintervallumokban
- Vektorkvantálás: a jellemzők hiperterének régióihoz rendelése a keretek jellemzővektorainak.





■ A hangjel feldolgozása ..

- A **beszédfelismerés** célja a beszélőktől független, de a beszédre jellemző paraméterek kinyerése
- A **beszélőfelismerés** ezzel szemben éppen a beszélőkre jellemző paraméterek kinyerését jelenti.
- A szavak egyértelmű felismerése további **statisztikai**, illetve **valószínűségi** adatok felhasználását kívánja meg.
- A legjobb rendszerek a szavak több, mint 95%-át jól ismerik fel.
- A beszédfelismerő rendszerek statisztikai, valószínűségi adatait általában betanítással adják meg.

■ A legjobb beszédfeldolgozó rendszerek

- A SpeechStar programcsomag, benne a Dragon **NaturallySpeaking** beszédfelismerővel
- Az IBM **ViaVoice** programja
- Lemut & Hauspie **Voice Xpress** programja

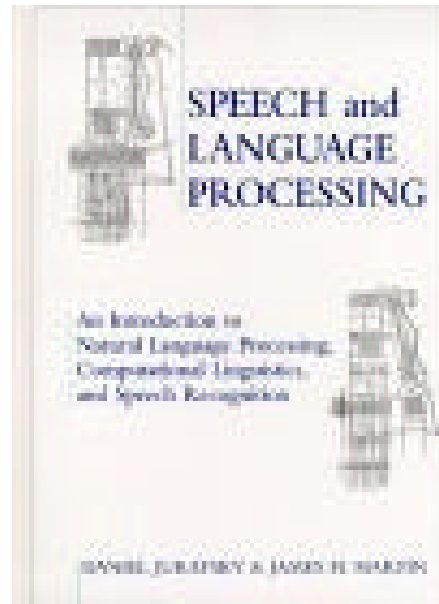


[Http://www.dragonsys.com/](http://www.dragonsys.com/)

<http://www-3.ibm.com/software/speech/desktop/w8-win.html>

■ Természetes nyelvek feldolgozása

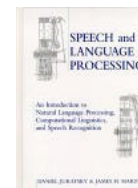
- Cél: Az emberi nyelv (írott formájának) megértése. Szükséges az MI eszközök és az ember közvetlen kapcsolatához. Idetartoznak a természetes nyelvű interfészek és a gépi fordítás is.



Dan Jurafsky, James H. Martin, Keith Vander Linden, Daniel Jurafsky: Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition

■ Természetes nyelvek feldolgozása ..

- **Példák: ELIZA** (*Joseph Weizenbaum*, MIT, 1966)
 - Ember - gép párbeszédet valósított meg. A gép egy pszichológust utánoz, aki megpróbálja megérteni az ember problémáit. Valójában az ember válaszainak kulcsszavait érzékelve típusszöveget, **egyszerű mintaillesztésen alapuló szövegrész-cserét alkalmazott**. Hasonló programot írt *K.M.Colby* az 1960-as évek közepén. Valódi célja a Turing teszt teljesítésének megpróbálása volt, sikertelenül.



A terminálokon folytatott párbeszéd például így festett:



Orvos: Hogyan tudnék segíteni?

Páciens: Nem tudom.

Orvos: Próbáljon mondani róla valamit!

Páciens: Nem tudok.

Orvos: Ki tudná ezt fejteni részletesebben?

Páciens: Beteg vagyok és fáradt.

Orvos: Van elképzelése, hogy miért?

Páciens: Igen, öreg vagyok.

Orvos: Folytassa, kérem!

Páciens: Úgy érzem, elvesztettem minden érdeklődésemet.

Orvos: Mondjon többet ezekről az érzésekről!

Páciens: Van arról is elképzelésem, miért érzek így.

Orvos: Biztosan van.

■ Természetes nyelvek feldolgozása ..

- **Példák: SHRDLU** (*Terry Winograd, 1972*)

- Egy mértani testekből (kockák, gúlák, stb.) álló mikrovilágot hozott létre, melynek elemeit egy robot az ember írásos parancsainak megfelelően manipulálhatta.

pl.:

Ember: "Keress egy hasábot, amelyik magasabb, mint az, amelyiket a dobozba raktál !"

Robot: "Rendben."

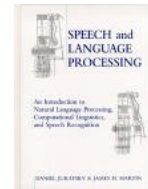
Ember: "Megérintettél egy gúlát, mielőtt a zöld hasábot a kis kockára tetted ?"

Robot: "Igen, a piros színűt."

.

.

A rendszernek a kockák mikrovilágáról egy belső leképezése volt, és azt felhasználva, látszólag megértette az ember mondatait.



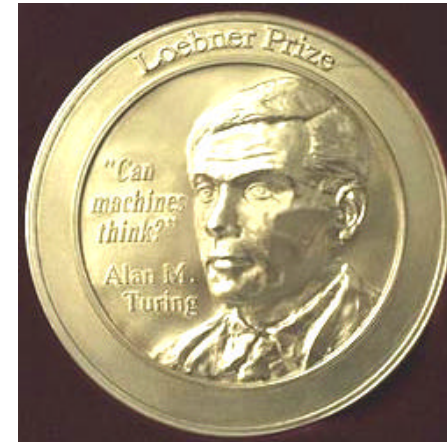
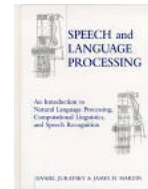
■ Természetes nyelvek feldolgozása ..

- **A Loebner - díj**

- Hugh Loebner 1990-ben 100 000\$-os díjat és egy arany medált tűzött ki annak a nyertesnek, akinek a beszélgetőprogramja elsőként teljesíti a Turing tesztet. Egy 2000\$-os kisebb díjat és egy bronz medált minden évben elnyer az a program amely a legszínvonalasabb párbeszédet folytatja emberrel.

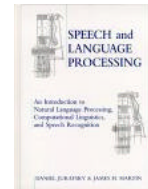
- **A nyertes 2001-ben (is) Richard Wallace amerikai programozó volt. Beszélgető-robotja kipróbálható az Interneten.**

- **További chat-robotok találhatóak a <http://alice.sunlitsurf.com/live.html> címen.**



■ Számítógépes fordítás ..

- A természetes nyelvek feldolgozása nagy jelentőségű a számítógépes fordítás, tolmácsolás megoldásához is. Az emberi nyelv összetettsége miatt a megoldást csak az emberi intelligenciával, tudással összemérhető képességű géptől várhatjuk a jövőben. Szűk területeken bizalomkeltő részeredmények vannak.
- Néhány on-line elérhető fordítóprogram címe:
<http://babelfish.altavista.com>
<http://www.babylon.com>
<http://www.freetranslation.com>
A hazai **Morphologic** cég is elkészítette fordítóprogramját, melynek előnye, hogy a magyar nyelvet is ismeri.
- A természetes nyelvek megértésének problematikájára jól rávilágít egy mondat értelmezésének 4 szintje:
 - » Szintaktikai
 - » Szemantikai
 - » Pragmatikus
 - » Intencionális.

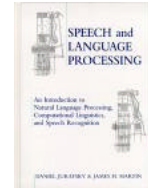


■ Számítógépes fordítás ..

- A mondat jelentésének négy szintje

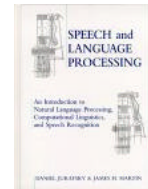
Az egyes szintek tartalmát legjobban egy példa kapcsán mutathatjuk be:

- *Éva: Tudod, hogy Viktor ugyanúgy dohányzott, mint te?*
 - *Imre: Nem. Miért, mi van vele?*
 - *Éva: Tüdőrák. Feldobta a bocskorát.*
 - *Imre: Szomorúan hallom.*
- Figyeljük azt a mondatot: "Feldobta a bocskorát."
 - Szintaktikailag (formailag) egy múlt idejű állítmányt és egy tárgyat figyelhetünk meg.
 - Szemantikailag (tartalmilag) azt jelenti: Felhajította a lábbelijét.
 - Pragmatikus (valóságos) jelentése: Meghalt.
 - A mondat intencionális (szándékolt) jelentése, célja a szöveggörnyezettel együtt Imre figyelmét felhívni, hogy ne dohányozzon annyit.
 - Az ELIZA program gyakorlatilag csak a szintaktikai szinten értette a páciens válaszait.



■ Számítógépes fordítás ..

- A *gépi fordítás* szintjei a fordítás *minősége* szerint a következők:
 - tájékozódó fordítás (information acquisition)
 - tényszerű közlésekre vonatkozó fordítás (denotative translation)
 - igényes fordítás (connotative translations).
- A gépi fordítás az *automatizáltság szintje* szerint lehet:
 - teljesen automatikus (fully automatic machine translation)
 - emberi segítséggel készülő (human assisted machine translation)
 - gépi segítséggel készülő (machine assisted human translation)
- Természetes nyelvű interfészek
 - A *természetes nyelvű interfészek* különféle számítógépes alkalmazások kezelését könnyítik meg azáltal, hogy a kezelő a megszokott szavakkal, mondatokkal kommunikálhat a szoftverrel. Természetes nyelvű interfészeket fejlesztettek ki: jól strukturált adatbázisokhoz, szimulációs modellekhez, szakértői rendszerekhez, helybiztosító rendszerekhez és szöveges adatbázisokhoz.



■ Korlátozás kielégítés

- A korlátozás kielégítési feladat a benne szereplő változók értékeit korlátozza. A korlátok megadhatók az értékek felsorolásával, explicit módon, vagy egy kifejezéssel, implicit módon. A változók által felvehető értékek száma véges.
- A feladat megoldása alatt a változók olyan értékhalmozát értjük, melyek kielégítik az összes korlátozást.
- Megelégszünk egyetlen megoldással, bár gyakran több megoldás is lehetséges. Egyes esetekben az összes megoldást meg kell találnunk.
- A korlátozás kielégítés kedvenc feladata volt a 8 vezér probléma: úgy helyezzünk el a sakktáblán 8 vezért, hogy ne üssék egymást.
- A megoldó módszerek egyike a lehetőségtér állapotait tartalmazó fagráf „mélységben először” technikával történő bejárása.
- A megoldáskeresés gyorsítható a korlátozás propagáló, azaz a korlátokon kívül eső, megoldást biztosan nem adó pontok kizárásával.

■ Cselekvési tervek generálása



- A cselekvéstervezés lényege hatékony célirányos tevékenységsorozat generálása valamilyen, a világban felmerülő feladat megoldására.
- A megoldás meghatározására általános, azaz a problémától független valamint alkalmazás-specifikus módszerek közül választhatunk.
- Megoldási módszerek:
 - keresés - nehezíti a nagy elágazási tényező, s a talált megoldások a cselekvések egyszerű szekvenciái lehetnek csak.
 - Szituációkalkulus - nehezen irányítható és könnyen adódnak nem megfelelő lépések is. (A szituációkalkulus az elsőrendű logika módszerét alkalmazza a világ egy adott állapotára, azaz egy szituációra.)
- A cselekvési terv a két módszer együttes alkalmazásával áll elő, a finomító tervezési szakasz előre/hátra láncoló technikájánál a keresés jut szerephez.
- Végeredményként egy olyan cselekvéseggyüttes jön létre, amely végrehajtható és a világot a megadott korlátokat kielégítő új állapotba viszi át.

■ Szakértőrendszerek



- Egy szakértőrendszer olyan eszköz, amely probléma specifikus ismeret megértésére képes és intelligensen használja a tématerület ismeretanyagát egy tevékenység különböző megvalósítási útjainak felvetéséhez. A szakértőrendszerek nem csak az ismeretátadás technikáit alkalmazzák, hanem analitikus, elemző eszközöket is az ismeret kiértékelésére, valamint tanulási technikákat.

Példák olyan területekre, ahol szakértőrendszert alkalmaznak:

- Repülés: repülőgépmotor-diagnosztika: helikopter javítás, Pl.: NAVEX,
- Mezőgazdaság: Almáskertek gondozása (POMME),
- Kémia: Kémiai reakciók tervezése (SYNCHEM),
Szerkezetértelmezés (DENDRAL),
- Számítógépek és kommunikáció: VMS dump fájlok elemzése rendszerkiakadás után (CDX),
- Oktatás: Tervezők oktatása konstrukciós tervezés ellenőrzésére (DECGUIDE),
- Vállalatvezetés: Üzlet hatékonyságelemzése (GURU),
- Egészségügy: Fertőző betegségek diagnózisa (MYCIN),
- Mérnöki technika, gépészet: Motoralkatrészek tervezése, stb.

■ Szakértőrendszerek: a DENDRAL



- Ma is használt szakértőrendszer, több, mint 50 tudományos cikket írtak a vele elért felfedezésekről.
- **Célja:** Hipotézis felállítása egy vegyület lehetséges molekuláris felépítésére, atomszerkezetére.
- Ha egy vegyész egy ismeretlen vegyülettel találkozik, első feladata, hogy meghatározza az összetevő atomokat és azok relatív arányát. Ehhez analitikai tesztekre és kísérletekre van szüksége. Az egyik gyakran alkalmazott műszer a tömegspektrométer. A DENDRAL is felhasználja a vegyület tömegspektrumát.
- Egy vegyület kémiai képletét egyszerű megtalálni, de **szerkezeti képletének** meghatározása speciális tudást igényel.

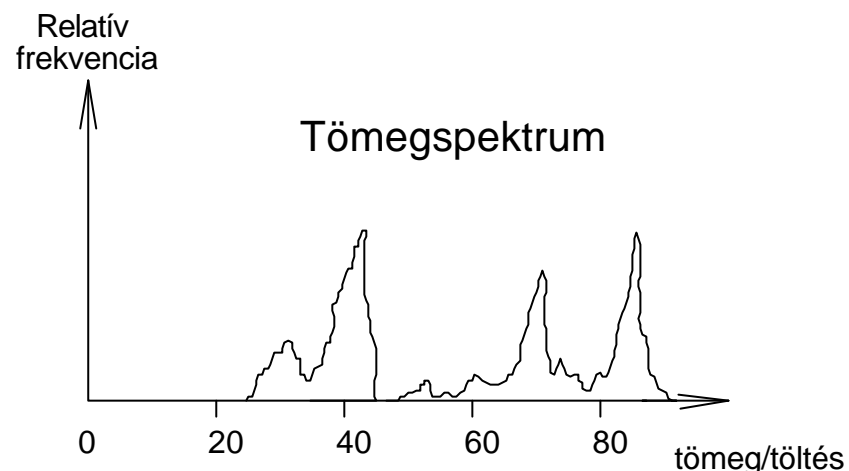
■ Szakértőrendszerek: a DENDRAL ..



- **A DENDRAL három részből áll:**
 - **Szerkezetgeneráló** rész, az összegképletből a lehetséges szerkezeti képletet generálja;
 - **Spektrumjósoló**, a szerkezeti képletből a megfelelő jósolt tömegspektrumot származtatja, és összehasonlítja az aktuális adatokkal;
 - **Szerkezetjósoló**, amely résszerkezeteket származtat le az adatokból és lehetetlen szerkezeteket kizár. Ez a rész felel meg a szakértő tudásának (1. + 2. elvileg elég, de kombinatorikus robbanás miatt nem lehet az összes szerkezetet generálni és vizsgálni. pl.: $C_6H_{13}NO_4$ kb.10000 lehetséges szerkezettel rendelkezik.).



■ Szakértőrendszerek: a DENDRAL ..



- i. szabály:** ha létezik egy magas csúcs a 71 tömeg/töltés helyen és
ha létezik egy magas csúcs a 43 tömeg/töltés helyen és
ha létezik egy magas csúcs a 86 tömeg/töltés helyen és
ha van néhány csúcs az 58 tömeg/töltés helyen
akkor az N-propil-ke-ton 3 szerkezet jelenlétének feltételezhető.

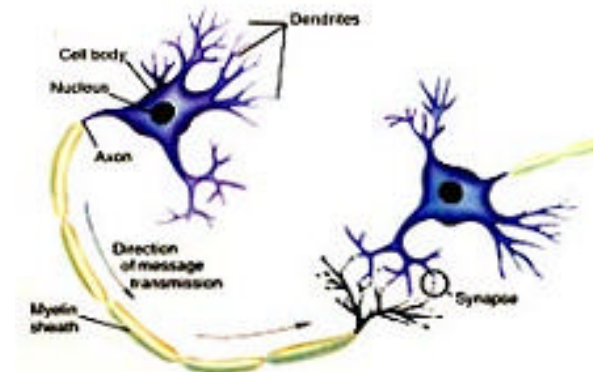


■ Mesterséges neurális hálózatok

- Az MI egyik nem új, de egyre nagyobb jelentőségű területe az emberi agy neuronjainak modellezésével létrehozott neurális hálók kutatása, ill. alkalmazása.
- A neurális hálózatot szokták kapcsolati modellnek (connectionist model) nevezni a működésben domináló kapcsolatok és nagyon erős párhuzamos működés miatt.

3 fő részük:

- a neuronok
- a hálózati topológia (kapcsolódások)
- a tanulási algoritmus.



1.

1. <http://ei.cs.vt.edu/~history/NEURLNET.HTML>

■ Mesterséges neurális hálózatok ..

képességei

- mintafelismerés
- általánosítás
- trendek megjóslása
- viselkedés, kimenetel megjóslása
- kiértékelés
- nem pontos adatok elfogadása
- szűrés
- gyors működés
- szövevényes viszonylatok felfogása
- előnyös leegyszerűsítés
- optimalizáló képesség
- hatalmas adatmennyiség elemzése
- extrapolálás

alkalmazására példa [17]

tengeralattjárók felismerése
sonar -jelek alapján

a valós állapot felbecsülése
döntés részvényvásárlásról, vagy
eladásról

a műtét kimenetelének
megjóslása

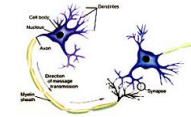
kölcsönigények elfogadása,
elutasítása

optikai karakterfelismerés
videojelek zavarmentesítése
robotkar vezérlése

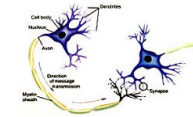
gyógyászati "szakértőrendszer"
vezérlés az űrben

repülőjáratok ütemezése

biztosítási igények összevetése
termelési problémák diagnózisa.



■ Mesterséges neurális hálózatok ..



Egy konkrét neuronháló alkalmazás: Alkatrészcsaládok kialakítása [18]

- Adott m gép és n alkatrész, valamint egy $m \times n$ méretű mátrix, melyben 1 áll az ij helyen, ha az i . gépen meg kell munkálni a j . alkatrészt.
- A neuronhálónak a feladata egy ilyen mátrixhoz, mint input mintához egy olyan output mintát rendelni, melynek elemei megfelelnek egy-egy alkatrészcsaládnak.
- Egy család alkatrészeit az jellemzi, hogy ugyanazon a néhány gépen - azaz egy gépcsoporton - minden műveletük elkészíthető.

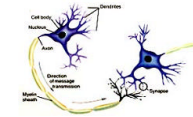
		Alkatrészek						
		1	2	3	4	5	6	7
Gépek	1		1		1	1	1	
	2	1		1				
	3	1	1					1
	4		1		1		1	
	5	1						1

Az alkatrészcsaládok

1. család: 1,3,7
2. család: 2,4,5,6

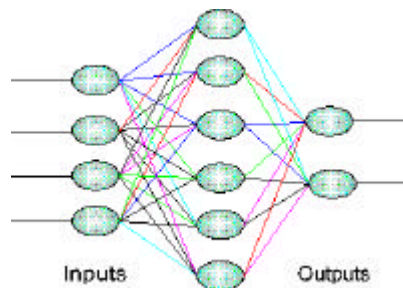
Műveleti mátrix

■ Mesterséges neurális hálózatok ..



Egy konkrét neuronháló alkalmazás: Alkatrészcsaládok kialakítása ..


- A megoldás kihasználja a versengő neuronháló azon tulajdonságát, hogy a neuronokat képesek osztályokba rendezni, ezzel gyakorlatilag megoldva a *csoporttechnológia* (Group Technology) alapfeladatát.
- Előnyök:
 - könnyű alkalmazhatóság,
 - új alkatrész könnyen bevonható,
 - nagy méretű feladatok időigénye nem nő exponenciálisan.



<http://www.e-orthopaedics.com/sakura/neusak.htm> Ez a háló nem versengő háló.

■ Adatbányászat

Az adatbányászat az adatbázis összefüggések, mintázatok után segít a szolgáltató vállalatoknak vevőket. Az adatbányászat kifejezést olyan szoftverek meghatározására, adatokat. Az igazi adatbányászati prezentálását változtatják meg, hanem korábban nem ismert összefüggéseket is feltárnak az adatok között.



alkalmazások egyik területe amely rejtett kutat nagy adathalmazokban. Például, megtalálni az azonos érdeklődésű rendszerint helytelenül használják amelyek új módon prezentálják az szoftverek nem csak az adatok

- Az adatbányászat csak pár évtizedre visszatekintő múlttal bíró, dinamikusan fejlődő új kutatási irányzat. Célja az adatbázisokban található ismeretekből a **tudás kinyerése**, ezáltal konvergál a mesterséges intelligencia tudománya felé.
- Az adatbányászat kezdetben az általánosan alkalmazott statisztikai módszerek egyszerűsítésére és automatizálására fektette a hangsúlyt. Ez az irányultság az évek folyamán megváltozott. Sok új adatbányászó algoritmus és eszköz keletkezett. Ezek összevetéséhez komoly statisztikai felkészültség kell.
- Egyre több intelligenciát építettek be a programokba a mesterséges intelligencia többnyire jól ismert módszereit alkalmazva.

■ Adatbányászat

Alkalmazott módszerek:

- Felügyelt
 - Regressziós technikák
 - Legközelebbi szomszéd módszere
 - Mesterséges neurális hálók
 - Következtető (indukciós) szabályok
 - Döntési fák
- Felügyelet nélküli
 - Klaszterezés
 - Önszervező neurális hálók



Regression Models



K Nearest Neighbor



Neural Networks



Decision Trees



Self Organized Maps



Text Mining



Sampling



Outlier Filtering



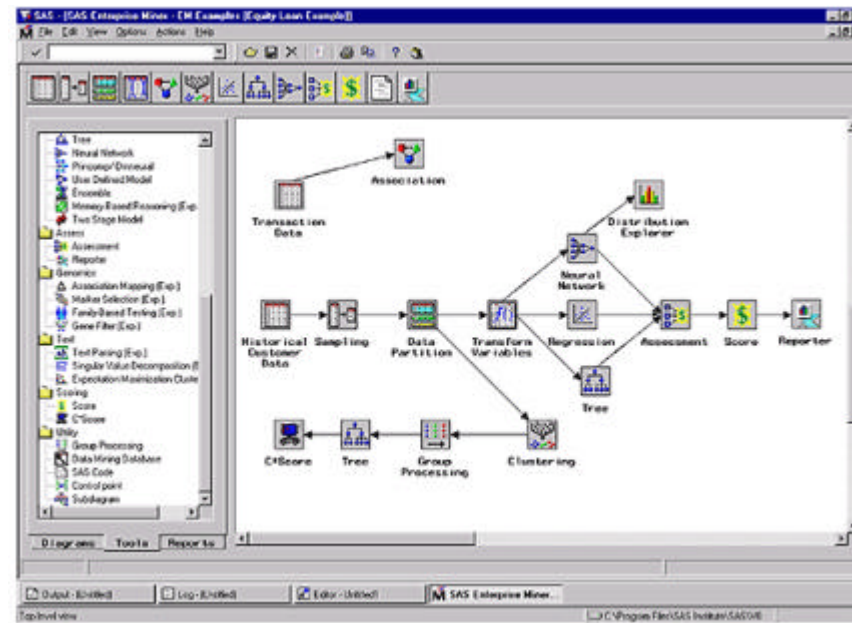
Assessment

■ Adatbányászat

Kiemelkedő alkalmazásszállítók:

- SAS
- SPSS
- Oracle
- Angoss
- HNC
- Unica.

SPSS Clementine



■ Az ötödik generációs számítógép projekt

- Az MI és a szoftverfejlesztés területén végzett kutatások előrehaladásával világosabbá váltak az okok, hogy eddig (1970) miért nem tudtak megbirkózni a számítógépek az olyan feladatokkal, mint a mesterséges intelligencia tárgykörébe tartozó feladatok, ill. a párhuzamos feldolgozást igénylő feladatok.
- Maguk a feldolgozási módszerek is világosabbak lettek. Felmerült annak a lehetősége, hogy további kutatásukkal olyan számítógép építhető, amely a felhasználó számára már természetesebben megközelíthető.
- A japánok hirdették meg az 5. generációs számítógép létrehozására irányuló projektet. Ebben célul tűzték ki, hogy az 5. generációs számítógépnek **az átlagos emberi tudást kell tárolnia.**

■ Az ötödik generációs számítógép projekt ..

- Az eredmények amelyekre támaszkodtak:
 - VLSI technológia, mint lehetőség: szilícium alapú logikai kapuk néhány száz picosecundum alatti jeltovábbítással (1 picosec = 1 sec / 1000millió; a fény 100 picosec alatt 3 cm-t tesz meg.);
 - szupravezetők;
 - optikai technológia (képfeldolgozáshoz természetes);
 - párhuzamos feldolgozás, pl. a *J.B.Dennis* által kidolgozott adatvezérelt működés, mely lehet:
 - adatmeghajtású (ha van input, elvégzi a számítást),
 - igénymeghajtású (a számítást akkor végzi el, ha már szükség van az outputra);
 - párhuzamos számítási algoritmusok, -programozás;
 - tudásszemléltető és feldolgozó nyelvek: LISP, PROLOG.

■ Az ötödik generációs számítógép projekt ..

1. természetes nyelv, nyelvi és grafikai alkalmazás 2. magasszintű elemzőnyelv 3. tudásbázist kezelő rendszer 4. intelligens interfészkezelő rendszer 5. belső nyelv 6. problémamegoldó és következtető rendszer 7. külső interfész az alap szoftverrendszerhez

8. tudásbázist kezelő rendszer 9. problémamegoldó és következtető rendszer 10. intelligens interfészrendszer 11. alap szoftverrendszer

12. tudásbázisú gép 13. relációs algebra 14. relációs adatbázisrendszer 15. problémamegoldó és következtetőgépek 16. predikátumkalkulus típusú nyelv 17. absztrakt adattípust támogató rendszer 18. párhuzamos adatáramlásos rendszer 19. fejlett Neumann-rendszer 20. intelligens interfész gép 21. hardverrendszer 22. elosztott funkciójú hálózati rendszer

23. VLSI architektúra.

