

# Vypracované otázky z UIR

1 Vysvětlete pojmy „inteligence“, „umělá inteligence“, „počítačová inteligence“ a jaké komponenty můžeme v počítačové inteligenci identifikovat :

- a) **Inteligence** – schopnost člověka abstraktně a rozumně přemýšlet a odvozovat účelná jednání
- b) **Umělá inteligence** – schopnost některých programů napodobovat inteligenci člověka tím, že vykonávají činnosti spojené s člověkem jako např. uvažování, učení a sebezdokonalování.
- c) **Počítačová inteligence** – obor umělé inteligence, aplikace algoritmů biologického světa do informatiky

**Komponenty v počítačové inteligenci** – fuzzy logika (mlhavá logika, operuje se všemi hodnotami z  $(0; 1)$ ), neuronové sítě, evoluční algoritmy

2 **Informatická definice umělé inteligence (Sowa):**

Pod pojmem „umělá inteligence“ chápeme snahu člověka o vytvoření počítačových programů, které by se schopností rozhodovat se v předem neočekávaných situacích blížily uvažování a chování člověka a věrně jej napodobovaly.

3 **Specifikujte jednotlivé problémové oblasti umělé inteligence:**

- d) Řešení úloh, herní strategie
- e) Automatické dokazování – provádění důkazů na základě znalostí, které lze formalizovat a v jednoduché formě uchovávat v paměti
- f) Inteligentní přístupy do databází – nalezení dat s co nejmenší prodlevou, systém si dokáže vydedukovat odpověď
- g) Reprezentace znalostí a znalostní systémy
  - fakta a znalosti, které si uchovává/používá
  - výsledek je rozhodnutí stavem diagnózy
- h) Kombinatorické a rozvrhovací úlohy – plánování a rozvrhování činností
- i) Vnímání prostředí – analýza scény, využití rozpoznávání
- j) Porozumění přirozenému jazyku – porozumět pokynům člověka
- k) Robotika – autonomní a cílově orientované instrukce s reálným prostředím
- l) Automatické programování

4 **Vysvětlete blíže vztah data vs informace:**

- **Data** jsou to, co přístroje zachytí z prostředí (bez souvislostí) například soubory číslic. Fakta bez souvislostí.
- **Informace** jsou data interpretována v definovaných souvislostech. Pokud by datem
- Znalost je postup, jak získaná data interpretovat a využít

## 5 Co je úloha a jak lze úlohu definovat:

Úloha je vlastně zobrazení množiny výchozích stavů do množiny cílových stavů:  $X \rightarrow Y$   
Grafem ji pak můžeme reprezentovat jako posloupnost stavů.

Úlohou také můžeme označit trojici  $(X, Y, R)$ , kde  $X$  je množina výchozích stavů,  $Y$  množina cílových stavů a  $R$  je kompoziční operátor

## 6 Obecná definice úlohy a co je řešením úlohy

- Obecná definice viz otázka 5.
- Řešení úlohy je postup, kterým převedeme úlohu z některého výchozího stavu  $X$  do definovaného cílového stavu  $Y$ .

## 7 Vysvětlete pojmy „elementární operátor“ a „kompoziční operátor“:

- **Elementární operátor:** operace vedoucí k transformaci jednoho mezistavu úlohy do jiného
- **Kompoziční operátor:** „množina elementárních operátorů“. Pokud je více výchozích a cílových stavů, lze řešení úlohy vyjádřit jako:  $X \xrightarrow{R} Y$  kde  $R$  je kompoziční operátor.

## 8 Typy úloh a jednoznačnost jejich řešení:

Úlohou také můžeme označit trojici  $(X, Y, R)$ , kde  $X$  je množina výchozích stavů,  $Y$  množina cílových stavů a  $R$  je kompoziční operátor. Podle této trojice rozlišujeme:

1. **Abduktivní**  $(?, Y, R)$  - 1 exaktní řešení
2. **Deduktivní**  $(X, ?, R)$  - 1 exaktní řešení
3. **Induktivní**  $(X, Y, ?)$  - řešení je hypotéza, kterou musím zpětně dokázat dedukcí

## 9 Vytváření a prohledávání stromu řešení úloh

Strom řešení se vytváří metodou pokusů a omylů. Jeho současným prohledáváním se pak hledá takový kompoziční operátor  $R$ , který vyhovuje množinám stavů  $X$  a  $Y$ .

Procházení stromu:

- Deterministické - předvídatelné
- Náhodné
- Heuristické – zkusmé řešení

## 10 Metoda stavového prostoru a způsoby reprezentace úlohy:

Zobecnění úlohy jako posloupnost mezistavů a elementárních operátorů (viz otázka 5). Zobecněním pak dostáváme **metodu stavového prostoru**. Předpoklady:

1. Konečná množina stavů, ve kterých se úloha může nacházet:  $S = \{s_i\}$ , pro  $i = 0, 1, \dots, N$
2. Existuje alespoň jeden výchozí stav
3. Existuje konečná množina cílových stavů:  $G = \{g_j\}$ , pro  $j = 0, 1, \dots, M$  taková, že
4. Existuje konečná množina elementárních operátorů:  $R = \{r_l\}$ , pro  $l = 0, 1, \dots, L$  které převádějí úlohu ze stavu  $s_p$  do stavu  $s_q$ , pro  $p, q = 0, 1, \dots, N$

Stavový prostor je pak definován dvojicí  $(S, R)$  a konkrétním řešením úlohy je trojice  $(s_0, G, R_{koj})$  kde  $R_{koj}$  je kompoziční operátor, který převede úlohu z počátečního stavu  $s_0$  do cílového stavu  $s_j = g_k$

## 11 Obecná metoda reprezentace úlohy ve stavovém prostoru a hledání řešení úlohy

Z předchozí otázky víme, že stavový prostor je definován dvojicí  $(S, R)$  a konkrétním řešením úlohy je

trojice  $(s_0, G, R_{Koj})$  kde  $R_{Koj}$  je kompoziční operátor.

**Řešení úlohy** je pak viz Obrázek 1. Kde uzly jsou jednotlivé mezistavy úlohy a hrany jednotlivé elementární operátory.

### kompoziční operátor

$$R_{Koj} = r_1 r_2 r_3 \dots r_{r-1} r_r$$

takový, že

$$s_1 \leftarrow r_1 (s_0),$$

$$s_2 \leftarrow r_2 (s_1) = r_2 (r_1 (s_0)), \quad (\text{zapišeme } r_2 r_1 (s_0))$$

$\dots$ ,

$$s_r \leftarrow r_r (s_{r-1}) = r_r (r_{r-1} (\dots r_1 (s_0))).$$



Obrázek 1: Řešení úlohy (otázka 11)

## 12 Vysvětlete pojem „expanze uzlu“ a jak souvisí s hloubkou uzlu

Expanze uzlu je nalezení všech bezprostředních následovníků daného uzlu. Tzn., že na daný uzel aplikujeme všechny možné elementární operátory.

Hloubka uzlu = počet hran od kořene k uzlu.

## 13 Jak je definován strom řešení úlohy (obecná definice)

Je to orientovaný graf definovaný takto:

1. Graf má jen jeden **kořen**, ten reprezentuje výchozí stav.
2. U každého uzlu definujeme rekurzivně hloubku uzlu. Kořen má hloubku **0**. Pokud má uzel hloubku **d** pak každý jeho bezprostřední následovník má hloubku **d+1**.
3. List je uzel bez potomka a reprezentuje
  - **cílový stav úlohy**
  - stav, na který **nelze aplikovat** žádný z elementárních operátorů
  - stav, o kterém jsme usoudili, že je z nějakého důvodu **neperspektivní**.
4. Orientovaná hrana reprezentuje přechod úlohy z daného stavu do nového stavu

## 14 Co je „produkční systém“ a z čeho se skládá

Je to počítačový program složený primárně ze sady pravidel chování. Skládá se z:

1. Databáze úlohy obsahující fakta
2. Báze znalostí obsahující **produkční pravidla** ve tvaru:
3. Řídícího mechanismu který má za úkol:
  - a) Provést volbu, které pravidlo bude aplikováno
  - b) Vybrat fakta z databáze. Ta dosadit do podmínky zvoleného pravidla.

- c) Ukončit řešení, je-li splněna cílová podmínka
- 4. Množiny cílů, které mají být splněny

## 15 Co je „cílová podmínka produkčního systému“

Je to podmínka, při které se produkční systém zastaví.

1. **Explicitní** – odvozená z množiny cílů
2. **Implicitní** – nejde-li daný obsah databáze aplikovat žádné další produkční pravidlo

## 16 Co jsou neodvolatelné řídicí strategie? Uveďte příklad.

Strategie takové, které aplikují pravidlo (operátor) tak, že tento výběr operátoru **už nelze později změnit**. Používá se jen pro nejjednodušší úlohy například pro hlavolam „8“

Například **horolezcův algoritmus**.

## 17 Jaké znáte pokusné řídicí strategie

Strategie, které umožňují v případě potřeby se vrátit k určitému uzlu a na stav odpovídající uzlu, do něhož jsme se vrátili, aplikovat jiné pravidlo (operátor).

1. **Backtracking**
  - a) Stavový – pamatuje si předchozí stavy
  - b) Operátorový – pamatuje si jen aplikované operátory
2. **Hledání v grafu řešení úlohy**
  - a) Neinformované strategie (slepé)
  - b) Informované strategie (zpravidla heuristické)

## 18 Uveďte rozdíl mezi operátorovým a stavovým algoritmem navrzení

- **stavový** – uchovává v paměti předchozí stavy (náročný na paměť)
- **operátorový** – uchovává v paměti zásobník operací, ne stavů (málo paměti, ale hodně času)

## 19 Co je základem metod hledání v grafu a jaké mají vlastnosti

- Pamatují si celou doposud vygenerovanou část stromu.
- Základem je **expanze uzlu a ohodnocující funkce**.
- Paměťově náročné.
- Jsou použitelné účinné globální heuristiky = ohodnocující funkce.

## 20 Uveďte jednoduché řešení úlohy obchodního cestujícího (uveďte příklad):

Pro nakreslený ohodnocený graf postupuji v kreslení stromu řešení následovně:

1. Expanduji počáteční uzel. (najdu všechny jeho sousedy a hrany stromu ohodnotím hodnotou hran grafu)
2. Vyberu nejmenší hranu a uzel opět expanduji (nevracím se do uzlů, kde už jsem byl)
3. Pokud již uzel nemohu expandovat a nenašel jsem cílový stav, rekurzivně se vrátím.
4. Algoritmus se zastaví v případě, že jsem prošel všechny uzly.

## 21 Princip vytváření stromu řešení úlohy:

1. Expanduji uzel.

2. Vyberu stav kam se dostanu po hraně s nejmenším ohodnocením.
3. Opakuji kroky 1 a 2, dokud je co expandovat. Pokud dojdou k listu, který zároveň není cílovým stavem, rekurzivně se vracím a vybírám další hrany.
4. Pokud jsem po vrácení došel ke stejnému výsledku jako dříve. Zruším pointer na to řešení, do kterého jsem se dostal po hranách s větším ohodnocením.

## 22 Vlastnosti slepých strategií hledání řešení úlohy, ilustруйте na jednoduchých příkladech

1. Do hloubky
  - Každý podstrom řešení expanduji až do listu pak se popř. rekurzivně vrátím.
  - Ohodnocení uzlu = hloubka uzlu
2. Do šířky
  - Expanduji uzel, vyberu potomka s nejlepším ohodnocením ohodnocující funkce a expanduji dál. Tímto algoritmem se najde nejkratší cesta k cílovému uzlu.
  - Ohodnocení uzlu = hloubka uzlu

## 23 Co jsou „informované (cílené) strategie hledání řešení úlohy“:

Strategie, které hledají takovou heuristiku (ohodnocující funkci), která by našla řešení úlohy v relativně malém počtu vygenerovaných stavů.

Volí se heuristická funkce :  $F(n_i) = d(n_i) + w(n_i)$  kde  $d$  je délka cesty z kořene do uzlu  $n$  a  $w$  je celková diference (nesouhlasy ve stavu  $n$  od cílového)

Většinou se narazí na protichůdné požadavky, například když chceme **najít nejkratší cestu** a přitom chceme ještě **minimální nároky na prohledávání stromu**

## 24 Vysvětlete funkci A a A\* algoritmu:

Je to modifikace základního algoritmu hledání v grafu. Za optimální cestu se považuje cesta s minimální „cenou“ – definuje se ohodnocující funkce:  $F^x(n_i) = g^x(n_i) + h^x(n_i)$

kde  $g^*(n_i)$  je skutečná „cena“ cesty z  $n_0$  do  $n_i$  a  $h^*(n_i)$  je skutečná „cena“ z uzlu  $n_i$  do cílového uzlu  $n_g$

Protože ale neznáme cestu z kořene do cílového uzlu použijeme proto tento přepis:

$$F'(n_i) = g'(n_i) + h'(n_i) \quad \text{kde jsou použity pouze odhady cen jednotlivých cest.}$$

Pokud ještě navíc platí, že  $0 \leq h'(n_i) \leq h^*(n_i)$  mluvíme o tzv. A\* algoritmu.

## 25 Co je míněno „více informovaným algoritmem“

Algoritmus s ryze heuristickou funkcí  $h'(n_i)$ , která je těsnějším dolním odhadem funkce  $h^*(n_i)$  - vyžaduje přesnější heuristickou informaci.

## 26 Způsoby rozkladu úlohy na pod úlohy, algoritmus rozkladu

Jde o nalezení takového kompozičního operátoru, který postupně odstraní první, druhou, ..., m-tou diferenci úlohy. Tím dojde k rozkladu úlohy na  $m$  pod úloh.

Diferenci se rozumí nesouhlas v dílčích složkách výchozího a cílového stavu.

## 27 Konjunktivně-disjunktivní graf pro reprezentaci rozkladu úlohy na pod úlohy

Také AND/OR graf, znázorňuje symbolický **rozklad úlohy na pod úlohy**. Je to orientovaný acyklický graf bez listů se dvěma typy uzlů:

1. AND – k vyřešení je potřeba vyřešit obě dílčí pod úlohy. Značí se obloukovým spojením obou pod úloh
2. OR – k vyřešení je potřeba vyřešit jednu z dílčích pod úloh



## 28 Algoritmy hraní jednoduchých her, základní rozdělení a implementační techniky

Jedná se vlastně o algoritmy prohledávání stavového prostoru reprezentovaného stromem řešení úlohy.

1. Prohledávající celý stavový prostor do určité hloubky - nejčastější
2. Prohledávající jen „dobře vypadající“ části stavového prostoru
3. Cílově orientované algoritmy – druhé nejčastější. Jsou dobré ale jen na jednu specifickou věc.

Cílem jednoho hráče je maximalizovat ohodnocující funkci a cílem druhého je ji naopak minimalizovat.

Implementují se tedy dva algoritmy – **minimax** a **alfa-beta prořezávání**

## 29 Ohodnocování uzlů stromu řešení herní úlohy při hledání řešení algoritmem minimaxu

Ohodnocení uzlů většinou symbolizuje pravděpodobnost, s jakou lze dosáhnout vítězství. Proto při hledání vítězného tahu na **úrovni hráče vybíráme maximum** a na **úrovni protihráče minimum**. Postupuje se od nejhlubšího vygenerovaného uzlu – musí být vygenerovaný celý strom do určité hloubky

## 30 Hledání vítězného tahu úlohy algoritmem alfa-beta prořezávání

Při vytváření stavového prostoru se dál nerozvíjejí ty větve, které mají horší ohodnocení než aktuální nejlepší. Na dané úrovni stromu si pamatujeme nějaké maximum/minimum.

## 31 Druhy evolučních algoritmů

- **Genetické** – založené na teorii přirozeného výběru. Zvládají problémy adaptace a měnící se podmínky učení.
- **Umělé neuronové sítě** – napodobují činnost mozku. Tvořené z neuronů a synapsí, kterými se simuluje chování nervové soustavy.

## 32 Typy operací používané v genetických algoritmech

- **Reprodukce** – „kvalitní“ jedinci mají častější reprodukci, kdy potomkům předávají něco ze sebe. Žijí také déle.
- **Křížení** – zabraňuje vývoji stejných jedinců a vznikají jedinci s prohozenými vlastnostmi
- **Mutace** – taky zabraňuje vzniku stejných jedinců. Lepší adaptace v měnícím se prostředí. Neprobíhá tak často jako křížení.

## 33 Implementace evolučních strategií a genetických algoritmů

- Jedinec řešení (**chromozom**) je nejčastěji reprezentován nějakou **lineární posloupností symbolů** (nejčastěji bitovým řetězcem). Každý chromozom je **ohodnocen funkcí (fitness)**, která mu přiřadí kvalitu (kladné číslo).
- Které chromozomy se budou reprodukovat, probíhá (kvazi)náhodně pomocí **rulety** (rozhoduje podle pravděpodobnosti).
- Pokud jsou už chromozomy natolik překřížené, že by dalším křížením vznikli ti samí jedinci, na malé části populace se provede **mutace**.

*Z populace se (kvazi)náhodně vyberou dva chromozomy, které si křížením vymění opět (kvazi)náhodně vybranou část řetězců. Výsledné chromozomy se pak ještě podrobí mutaci, která překloupí náhodně zvolený(é) bit(y). Takto nově vytvořená dvojice se vrací do populace, kde vytěsňují dvojici kvazínáhodně vybraných chromozómů s malou silou.*

### 34 Použití genetického programování

- Dobře fungují pro NP-úplné problémy. U jednoduchých úloh je nemá smysl používat.
- Vytváření rozvrhu práce strojů v továrnách  
V teorii her a managementu  
Optimalizační problémy multimodálních funkcí  
Řízení robotů  
Návrhy rozpoznávacích systémů  
Simulace úloh umělého života

### 35 Stochastické optimalizační algoritmy

- Používají se jen pro mnoha parametrické funkce s divokým průběhem
- Založeny na mnohokrát opakované náhodné volbě počátečního řešení úlohy - algoritmus je pak ale čistě heuristický

Například: Hill Climbing, Generické algoritmy, Evoluční strategie, Simulované žihání, Tabu search

### 36 Algoritmus simulovaného žihání – definice, implementace, praktické využití

Máme systém popsaný n-rozměrným stavem  $\mathbf{x} = (x_1, x_2, \dots, x_n)$ , kde každému stavu přiřadí funkce  $f(\mathbf{x})$  nějaké ohodnocení  $y$ . Úlohou je pak najít takové  $\mathbf{x}$  ve kterém funkce  $f$  nabývá globálního minima.

Jestli bude nový stav  $\mathbf{x}'$  přijat, se rozhoduje opět na základě pravděpodobnosti, kdy je vygenerováno náhodné číslo a pokud je menší než pravděpodobnost záměny stavů  $P(\mathbf{x}' \rightarrow \mathbf{x})$  pak je nový stav akceptován.

Volba parametru  $T$  velmi ovlivňuje výslednou pravděpodobnost (na začátku velké a postupem času klesá). Když je velké, přijmou se skoro všechny nové stavy a naopak pokud je malé nebude se přijímat skoro žádný nový stav.

Využití je výhodné v případech NP-úplných problémů, kdy je nalezené řešení totožné s optimálním řešením nebo je mu velmi blízké. Řešení se získává v dobrém čase.

$$P(\text{přijmu horší hodnotu}) = 1/e^{\frac{-f}{T}}$$

Například: problém obchodního cestujícího.

### 37 Algoritmus zakázaného prohledávání (tabu search) – definice, implementace, využití

- Vylepšení horolezcova algoritmu kdy se směr největšího spádu funkce neodvozuje z gradientu ale prohledáváním okolí.
- Z náhodně vygenerovaného bodu se v nějakém náhodně vygenerovaném okolí najde minimum. To se pak použije jako střed pro další minimalizaci. Algoritmus se pustí několikrát pro různé počáteční body a zaznamenává se nejmenší minimum, které se pak použije jako výsledek.
- Do algoritmu je zabudována krátkodobá paměť (implementována frontou) kde jsou uchovány inverzní transformace k předchozím iteracím. Ty jsou zakázány (odtud „tabu“). Je to proto, aby se algoritmus necyklil.

Například: problém obchod. cestujícího, rozvržení práce strojů v továrnách, strategické umístění skladů, atd.

### 38 Způsoby reprezentace rozpoznávaných objektů

1. Prostřednictvím signálů
  - **Jednorozměrné** (zvuky)
  - **Dvourozměrné** (snímky scén)
  - **Vícerozměrné** (obecné)



2. Podle charakteru signálu

- **Akustické** (zvuky, hudba, řeč, )
- **Vizuální** (reálné, upravené, symbolické, stylizované, ...)
- **Taktilní** (obecně reprezentované n-rozměrnými signály)

### 39 Charakterizujte pojmy „rozpoznávání“ a „klasifikace“ – definice, rozdíly

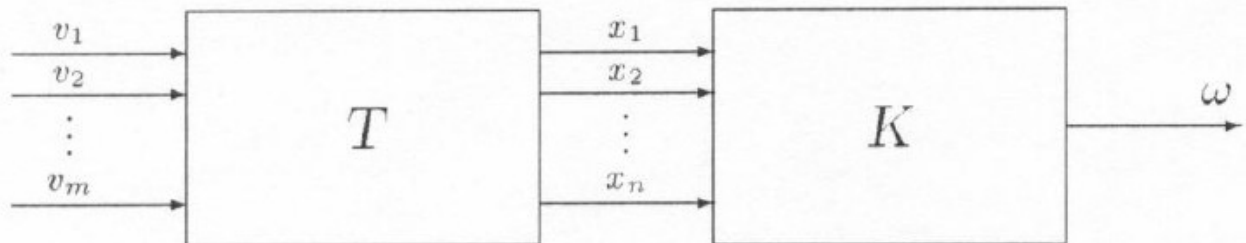
Rozpoznávání je zařazování objektů do tříd na základě jejich společných vlastností tak, že objekty vzájemně si podobné zařazujeme do stejných tříd.

- **Klasifikace** – zařazuje se do předem známého počtu tříd
- **Rozpoznávání** – počet tříd není dopředu znám a třídy identifikujeme až v průběhu vlastního rozpoznávání.

### 40 Shlukování

Zařazujeme objekty do tříd na základě podobnosti. Třídy nejsou předem známy.

### 41 Obecná klasifikační úloha



$T$  ... transformace vstupních charakteristik – vytvoření obrazu

$K$  ... klasifikátor

$v$  ... vektor vstupních charakteristik

$x$  ... obraz (symbolický popis) objektu

$\omega$  ... indikátor třídy

### 42 Co rozumíme pod pojmem „nastavení klasifikátoru“:

- trénování (učení) klasifikátoru.

Zvolení rozhodovacího pravidla (to rozhoduje, do jaké třídy bude obraz zařazen), které lze formálně zapsat skalární funkcí  $\omega = d(x)$  pokud ještě navíc zohledňuje nastavení klasifikátoru  $q$  lze napsat jako  $\omega = d(x, q)$

### 43 Metody učení klasifikátoru

- S učitelem – kdy klasifikátoru předložíme obrázky, u kterých známe třídu
- Bez učitele – kdy u obrázku neznáme třídu

### 44 Jak rozdělujeme vlastnosti rozpoznávaných objektů:

- Měřitelné a kvantifikovatelné (váha)
- Strukturní (s čím se učí)

### 45 Jak (podle čeho) dělíme metody rozpoznávání

Podle vlastností objektu hovoříme o příznakových nebo strukturních metodách rozpoznávání.

#### **46 Definujte pojem „obraz objektu“:**

Je to symbolický popis objektu.

- Tvořené příznaky (vektor příznaků)
- Strukturní

#### **47 Rozklad obrazového prostoru na třídy:**

Snaha o rozdělení obrazu do  $N$  nezávislých tříd ideálně disjunktních s velkou vzdáleností.

1. Disjunktní s velkou vzdáleností (ideální případ)
2. Disjunktní, kompaktní – geometricky blízké
3. Disjunktní, kompaktní, nelineárně oddělitelné nadplochou
4. Disjunktní, nekompaktní, lineárně oddělitelné
5. Disjunktní, kompaktní, vnořené, komplikovaně oddělitelné
6. Nekompaktní, prolínající se, velmi těžce oddělitelné

#### **48 Metody shlukování**

Rozlišujeme následující metody shlukování:

- Rozdělovací metody
- Hierarchické metody
- Metody založené na hustotě prvků
- Mřížkové metody
- Další metody (založené např. na neuronových sítích)

#### **49 Algoritmus k-means**

Tento algoritmus patří do rozdělovacích metod.

Princip: dělení prvků podle kritéria, podobnostní funkce (vzdálenost)

Shluk je reprezentován těžištěm objektů.

Popis algoritmu:

- Zadání počtu shluků  $k$  a množiny všech objektů
- Volba  $k$  výchozích „středů“ shluků
- Přiřazení všech objektů ke „středu“ shluku s nejmenší vzdáleností na základě podobnostní funkce (vzdálenosti)
- Výpočet nového „středu“ u každého shluku = „těžiště“ množiny objektů
- Návrat ke kroku 2
- Konec = nezměněna pozice žádného objektu

## 50 Jaké jsou základní typy dat?

- Intervalové proměnné
  - Př. Pozice, váha, výška, intenzita (jas), datum
  - Nutnost normalizace - obvykle interval [0; 1]
    - snaha nastavit všem proměnným stejnou váhu
- Binární proměnné
  - Pouze dva stavy – přítomnost vlastnosti (1) a nepřítomnost vlastnosti (0)
  - Př. Objekt složený z primitiv (čtverec, trojúhelník a kružnice)
    - Objekt č. 1 – 100
    - Objekt č. 2 – 101

## 51 Nejznámější metriky na měření vzdálenosti

- Eukleidovská vzdálenost  $\sqrt{\sum (a_i - b_i)^2}$
- Manhattan vzdálenost – stejná u všech cest  $|x_1 - y_1| + |x_2 - y_2| + \dots + |x_n - y_n|$
- Kosinová vzdálenosti – vzdálenost dvou vektorů je úhel který svírají

## 52 Evaluační metriky pro klasifikaci

Vyhodnocení přesnosti klasifikace. Udává se zpravidla v % nebo jako desetinné číslo.

### Chybovost (Error rate)

ER = E / ALL  
klasifikovaných vzorků

E = počet chyb, ALL = počet všech

### Přesnost (accuracy, klasifikace 1-1)

ACC = OK / ALL = 1 – ERR      OK = počet správně klasifikovaných vzorků

## 53 Evaluační metriky pro rozpoznávání

Vyhodnocení přesnosti rozpoznávání. Udává se zpravidla v % nebo jako desetinné číslo.

### Přesnost (accuracy, rozpoznávání 1-n)

ACC = (N – D – S – I) / N

- N = počet rozpoznávaných jednotek v sekvenci
- D = počet vynechaných jednotek, vynechaných jednotek
- S = počet změněných jednotek
- I = počet vložených jednotek

## Přesnost, úplnost, F-míra

Odhad / reál	1	0
1	TP	FP
0	TN	FN

TP = správně odhadnuté pozitivní příklady

TN = správně odhadnuté negativní příklady

FP = špatně odhadnuté negativní příklady,

FN = špatně odhadnuté pozitivní příklady

PŘENOST :  $AC = TP / (TP + FP)$

ÚPLNOST:  $CM = TP / (TP + FN)$

F-MÍRA:  $F = 2 \cdot AC \cdot CM / (AC + CM)$

### Konfúzní matice (matice záměn)

- matice, která obsahuje hodnoty přesnosti klasifikátoru s ohledem na všechny klasifikované prvky

### 54 Vysvětlete pojem matice záměn

- matice, která obsahuje hodnoty přesnosti klasifikátoru s ohledem na všechny klasifikované prvky

Například: klasifikace dokumentů, rozpoznání dialog. aktů

	A	B	C
A	88	4	12
B	8	65	4
C	11	15	33

### 55 Vlastnosti příznakových metod rozpoznávání:

Podle rozhodovacího pravidla nebo pokud zohledňuje nastavení klasifikátoru se rozhodne do jaké třídy rozpoznávaný vektor příznaků  $x$  náleží.

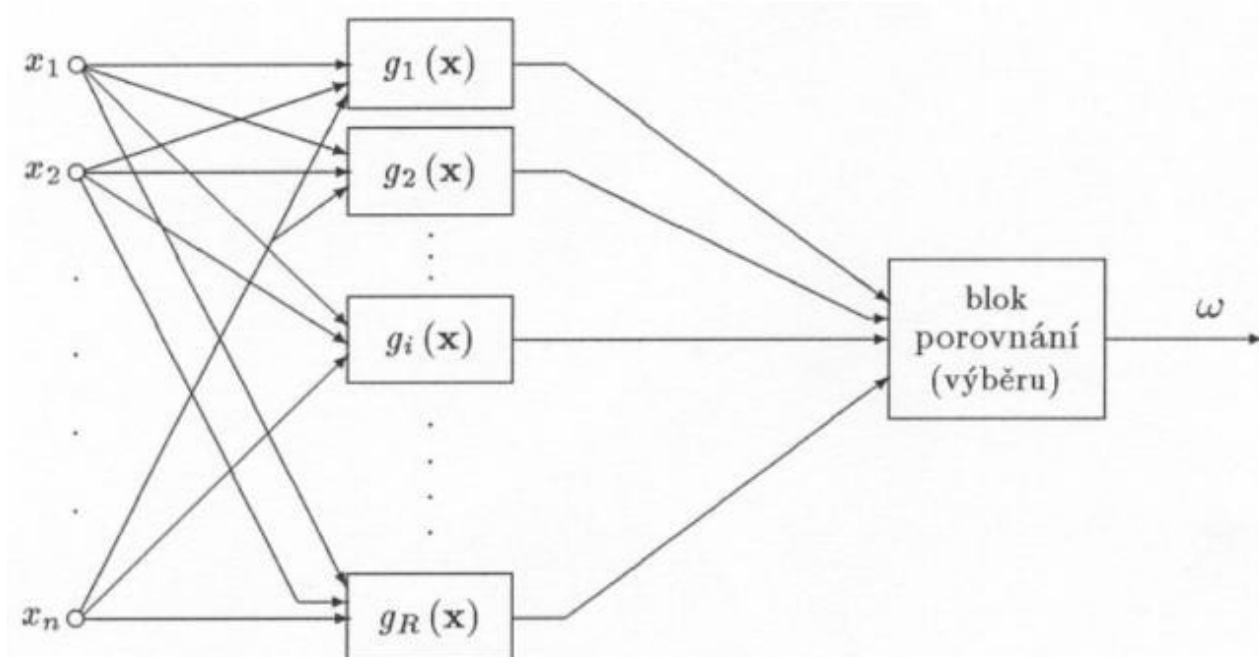
## 56 Klasifikátor s diskriminační funkcí – rozhodovací pravidlo, struktura klasifikátoru

Pro každou třídu definujeme takovou diskriminační funkci, aby všechny obrazy patřící do  $r$ -té klasifikační třídy platilo:

$$g_r(\mathbf{x}) > g_s(\mathbf{x}) \quad | \quad r \in \langle 1, R \rangle, \quad s = 1, \dots, R, \quad r \neq s.$$

Rozhodovací pravidlo pak nabývá tvaru

$$\omega_r = \max(g_s(\mathbf{x})), \quad s = 1, \dots, R$$



## 57 Klasifikátor na principu kritéria minimální vzdálenosti:

Lze použít v úlohách s oddělitelnými množinami obrazů. Ve fázi trénování se vytvoří vzorové obrazy (tzv. etalony) jednotlivých známých tříd – označíme  $e_s$

Neznámé obrazy se pak klasifikují pravidlem:  $\omega = \|e_r - x\| = \min \|e_s - x\|, s = 1, \dots, R$

## 58 Bayesův klasifikátor – rozhodovací pravidlo, návrh klasifikátoru

Pro úlohy s neodělitelnými (prolínajícími se) třídami obrazů, kde příslušnost ke třídě lze určit jen na základě pravděpodobností.

- lze nastavit tak, aby chyby byly minimální
- aposteriorní ppst příslušnosti třídy s identifikátorem  $\omega_r : P(\omega_s | x) = \frac{p(x | \omega_s) \cdot p(\omega_s)}{p(x)}$

Rozhodovací pravidlo je:  $\omega_r = \max(P(\omega_s | x)), \text{ pro } \forall s$ , kde  $\mathbf{x}$  je vektor příznaků.

## 59 Kritérium minimální chyby – ztrátová funkce, výpočet aposteriorních pravděpodobností

Pravděpodobnost, že obraz  $x$  patří do třídy  $\omega_r$  (aposteriorní ppst) je dána vztahem:

$$P(\omega_r|x) = \max(P(\omega_s|x)), \text{ pro } \forall s \quad \omega_r: P(\omega_s|x) = \frac{p(x|\omega_s) \cdot p(\omega_s)}{p(x)}$$

kde  $p(x)$  je histogram četností výskytu obrazů

$$\text{Ztrátová funkce } J(x) = \int \sum (\lambda(d(x, q)|\omega_s) \cdot p(x|\omega_s) \cdot P(\omega_s)) dx$$

## 60 Strukturní popis rozpoznávaných objektů, strukturní obrazy:

Obraz se rozloží na jednotlivá primitiva (jednotlivé části, ze kterých se skládá):

- Primitiva
- Vlastnosti primitiv
- Relace mezi primitivy
  - Prostorové
  - Časové
  - Funkční

Z jednotlivých primitiv se pak vytvoří strukturní popis (řetězec symbolů, relační struktura, graf).

## 61 Princip strukturního rozpoznávání objektů:

- rozhodnutí jestli je popis objektu (slovo) slovem jazyka některé z tříd (na základě gramatik)
- z toho se vytvoří strukturní popis
- výhody: jednoduchá reprezentace pro složité objekty, invariantní k natočení objektu

## 62 Úloha strukturního rozpoznávání objektů:

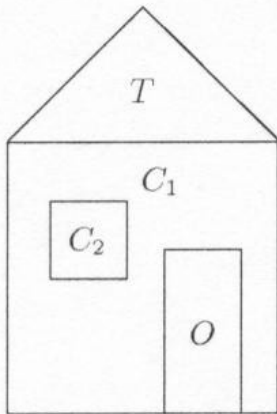
1. Trénování
  - a) Z trénovací množiny vytvořím primitiva a relace mezi nimi
  - b) Z primitiv udělám strukturální popis
  - c) Ze strukturálního popisu určím gramatiku pro každou třídu
2. Vlastní rozpoznávání
  - a) Z objektu vytvořím primitiva a relace
  - b) Vytvořím strukturální popis
  - c) Na základě gramatik třídy rozhodnu, kam patří
  - d) Přičítám obraz ke třídě

## 63 Postup vytváření strukturních popisů objektů (strukturních obrazů):

1. Nalézt všechna primitiva a přiřadit jim prvek nosiče struktury
2. Každému prvku přiřadit vlastnost (unární relaci) označenou „kódem“ (písmeno, číslo, ...)
3. Určit jednotlivé vztahy mezi primitivy (binární relace) -> vytvoří se relační struktura
4. Doplnit případné další informace číselné povahy (vytvoří se sémantický vektor)

# Vytváření popisné relační struktury (strukturního obrazu)

$\mathcal{M} \equiv \{T, O, C_1, C_2\}$



unární relace:

T ... je trojúhelník

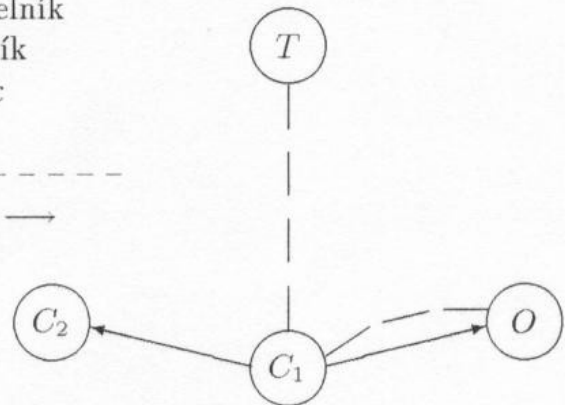
O ... je obdélník

C ... je čtverec

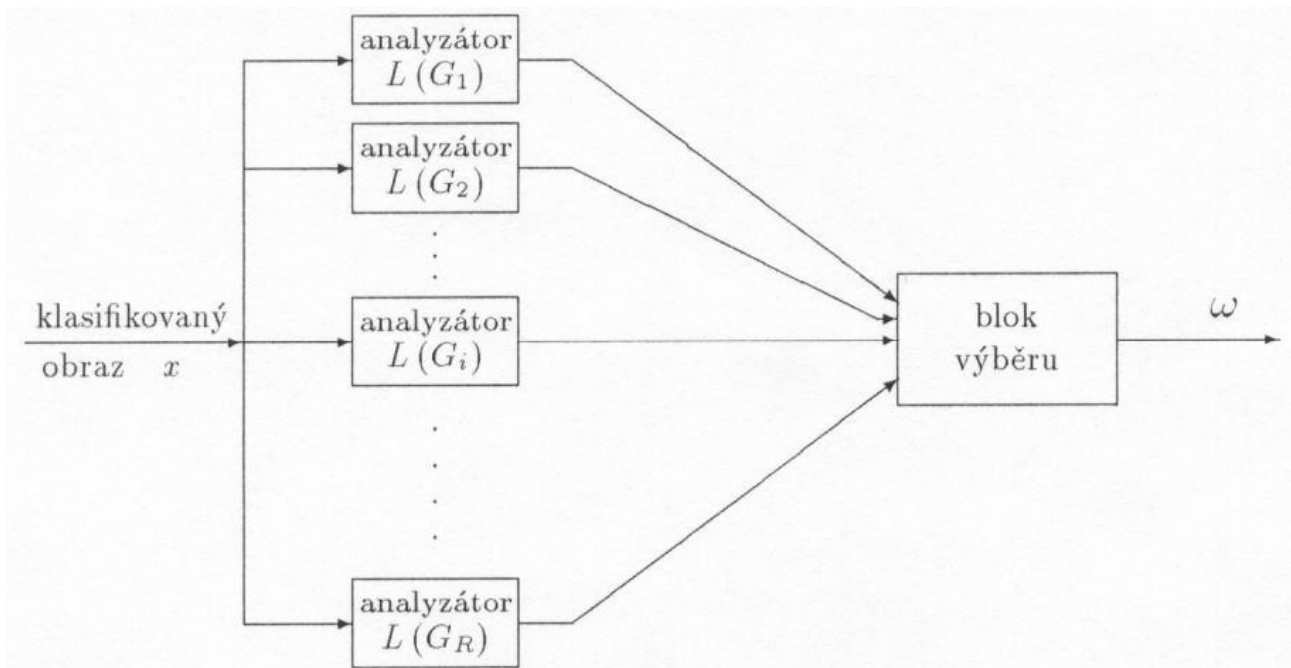
binární relace:

dotýká se ... - - - - -

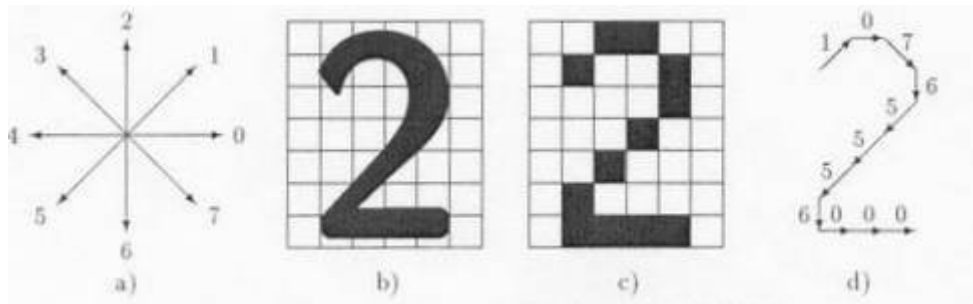
je uvnitř ... →



## 64 Struktura klasifikátoru pro strukturní rozpoznávání:



## 65 Princip extrakce primitiv Freemanovým kódem



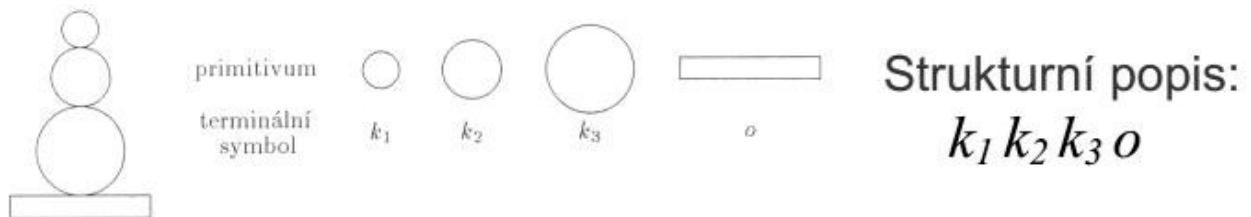
1. Směrová růžice Freemanova kodu
2. Překrytí objektu kódovacím rastrem
3. Repräsentace objektu v kódovacím rastru
4. Posloupnost směrů v kódovém rastru

Z toho dostáváme výsledný kód pro číslici 2: **1076556000**

## 66 Využití gramatiky pro strukturní popis objektů

Gramatika se využívá při shrnutí strukturního popisu do pravidel a při následném určování, do jaké třídy objekt patří.

## 67 Strukturní rozpoznávání s využitím sémantické informace



Obr. 5. 8: Obrázek typu "sněhulák" a volba primitiv

### „klasická“ regulární gramatika:

$$G_{sněh} = \langle V_{N_{sněh}}, V_{T_{sněh}}, S, R_{sněh} \rangle,$$

$$\text{kde } V_{N_{sněh}} \equiv \{S, A, B, C\},$$

$$V_{T_{sněh}} \equiv \{k_1, k_2, k_3, o\},$$

$$R_{sněh} \equiv \{S \rightarrow k_1 A, A \rightarrow k_2 B, B \rightarrow k_3 C, C \rightarrow o\}.$$

### Využijeme-li atributovou regulární gramatiku, dostaneme:

$$k_{(6)} k_{(8)} k_{(10)} o.$$

Třidu "sněhuláků" lze pak reprezentovat atributovou regulární gramatikou

$$G_{a_{sněh}} = \langle V_{N_{a_{sněh}}}, V_{T_{a_{sněh}}}, S, R_{a_{sněh}}, C_{a_{sněh}}, P_{a_{sněh}}, \mathbf{v} \rangle,$$

$$\text{kde } V_{N_{a_{sněh}}} \equiv \{S, A\}, V_{T_{a_{sněh}}} \equiv \{k, o\},$$

vektor  $\mathbf{v}$  obsahuje jediný parametr  $v_1$  označující poloměr koule a množina prepisovacích pravidel bude ve tvaru

$$R_{a_{sněh}}: \begin{array}{lll} & r_i & C_{a_{sněh}} & P_{a_{sněh}} \\ S \rightarrow & k A & & v_1 = 0 \\ A \rightarrow & k A & par \geq v_1 & v_1 = par \\ A \rightarrow & o & & \end{array}$$

Mám popsany objekt sadou pravidel nebo gramatikou a vyhledávám ho v obrázku. Využívá se tedy sémantické informace o objektu.



## 68 Proč je důležitý výběr příznaků v úloze klasifikace (rozpoznávání)?

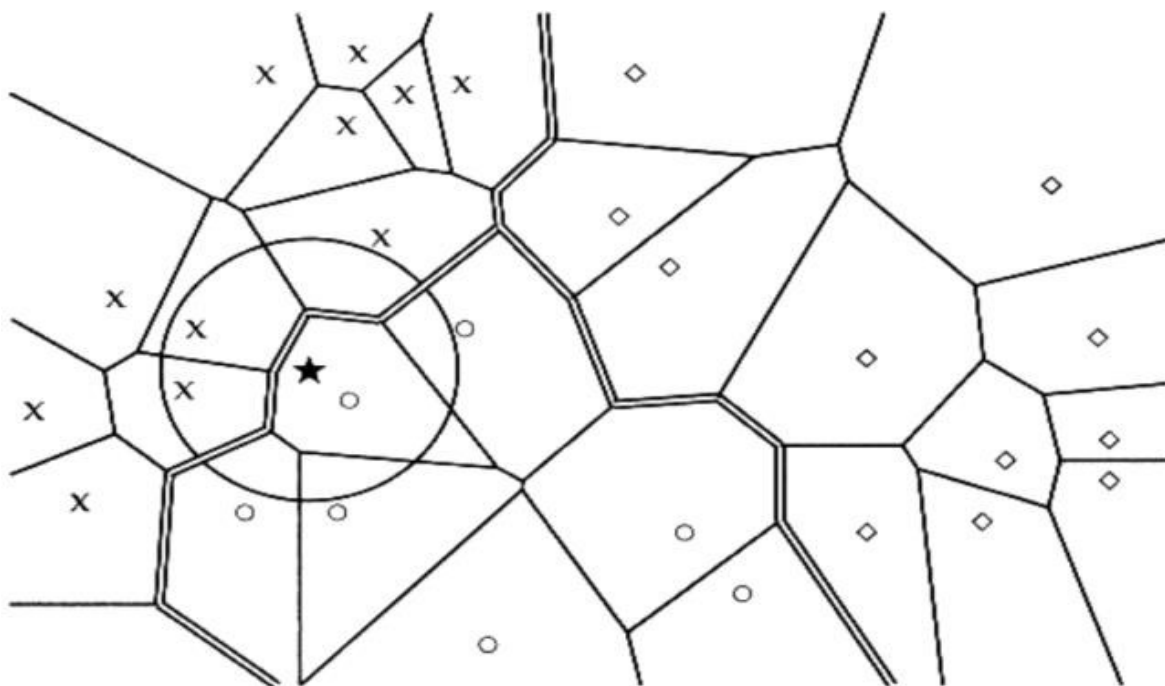
- Cena měření, výpočtu a použití (pro klasifikaci) všech příznaků
- Nevhodně zvolené příznaky (tj. i vyšší počet) – snížení ACC (přesnosti)

## 69 Základní metody pro výběr příznaků

- náhodný výběr - shora dolů / zdola nahoru; postupné ubírání/přidávání příznaků; ověření klasifikátorem
- frekvence dokumentu
  - jak často se term  $t_i$  vyskytuje v dokumentu  $d_i$
  - normalizuje se délkou dokumentu
- inverzní frekvence dokumentu – častá slova (např. „je“) jsou pro klasifikaci nedůležitá
- vzájemná informace
  - náhodné proměnné  $X, Y$  (znalost proměnné  $X$  sníží nejistotu  $Y$ )

## 70 Algoritmus k-nejbližších sousedů (k-NN)

- algoritmus strojového učení pro rozpoznávání vzorů
- metoda pro učení s učitelem
- fáze učení – trénovací množina tak, aby příznaky se střední hodnotou 0 a rozptylem 1
- fáze klasifikace – umístí prvek do N-rozměr. prostoru a najdu k-nejbližších sousedů
- Lokální hranice pro klasifikaci
- 1-NN - třída určena dle nejbližšího souseda
- k-NN - třída s max. počtem nejbližších sousedů (z celk. počtu k)
- Použití různých metrik



## 71 Klasifikační a regresní stromy (CART)

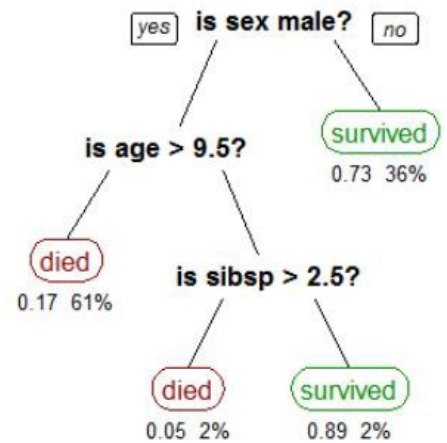
Popis vzájemných vztahů mezi pozorovanými veličinami pomocí stromu.

**Složení:** Kořen, uzly (větvení), listy (terminální uzly), hrany

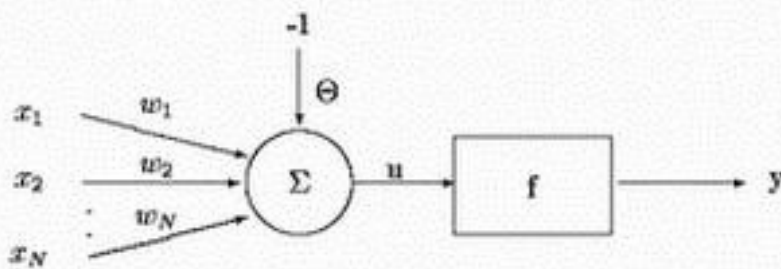
**Dělení:** Binární, Ternární

**Trénování:** Nastavení kritérií (vah) v uzlech stromu

**Testování:** Průchod stromem dle nastavených kritérií (vah)



## 72 Nakreslete model umělého neuronu a specifikujte jeho komponenty



$$y = f\left(\sum_{i=1}^N w_i x_i - \Theta\right)$$

$y$  – výstup neuronu

$u$  – vnitřní potenciál neuronu

$w_i$  – váhy neuronu

$x_i$  – vstupy neuronu

$\Theta$  – práh neuronu

$f$  – neuronová aktivační funkce

## 73 Co je to neuronová aktivační funkce a jaké jednoduché funkce používáme

Převádí vstup neuronu na výstup.

Sigmoidální funkce

$$f_s(u) = \frac{1}{1+e^{-u}}$$

Hyperbolický tangens

$$f_h(u) = \tanh(u)$$

Znaménková funkce

$$f_z(u) = \text{sgn}(u)$$

Heavisideova funkce

$$f_H(u) = \begin{cases} 1 & \text{pro } u > 0 \\ 0 & \text{pro } u \leq 0 \end{cases}$$

## 74 Způsoby (typy) učení umělých neuronových sítí

Cílem určení je nastavit váhy  $w_i$  tak aby síť vytvářela správnou odezvu.

- S učitelem – neuronová síť se učí srovnáváním aktuálního výstupu s výstupem požadovaným a nastavováním vah synapsí tak, aby se snížil rozdíl mezi oběma výstupy.
- Bez učitele – váhy sítě jsou nastaveny tak, aby byl výstup stejný u stejných, popřípadě podobných vektorů

## 75 Hebbův model učení umělých neuronových sítí

- učení bez učitele
- síla spojení dvou neuronů se zvětší, pokud jsou oba neurony aktivní. Síla spojení neuronů je přímo úměrná součinu jejich výstupních hodnot.

## 76 Typy umělých neuronových sítí

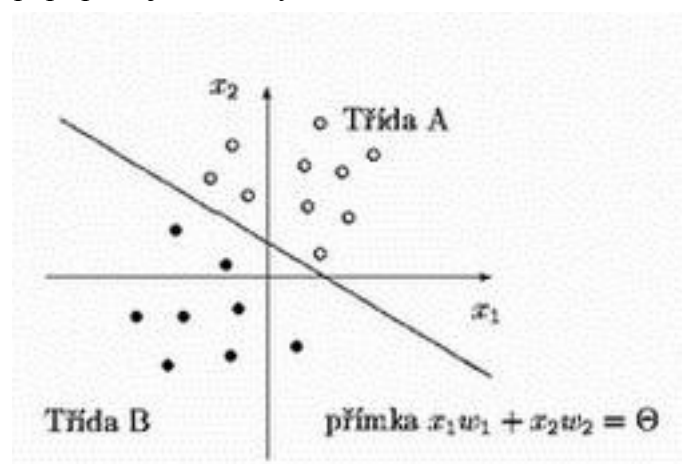
- pevné váhy
  - LAM, Hopfield, BAM
- učení s učitelem
  - Perceptron, MLP, TDNN
- učení bez učitele
  - ART1, ART2, Kohonenova síť, Neocogniton

## 77 Jednoduchý perceptron – struktura, způsob učení, použití

- Jednoduchý neuron nebo jednovrstvá síť s dopředným šířením (učení s učitelem)
- Neuronová aktivační funkce je : znaménková, popřípadě jednotkový skok
- **Použití:** lineárně separovatelné obrazy

### Trénování:

1. Počáteční inicializace vah ( $w_i$ ) a prahu ( $\theta$ ) neuronu (náhodně).
2. Přivedení vstupního vektoru a definice požadovaného výstupu.
3. Výpočet odezvy (výstupní hodnoty neuronu).
4. Přizpůsobení vah.
5. Opakuje se krok 2 – 4 pro všechny trénovací dvojice

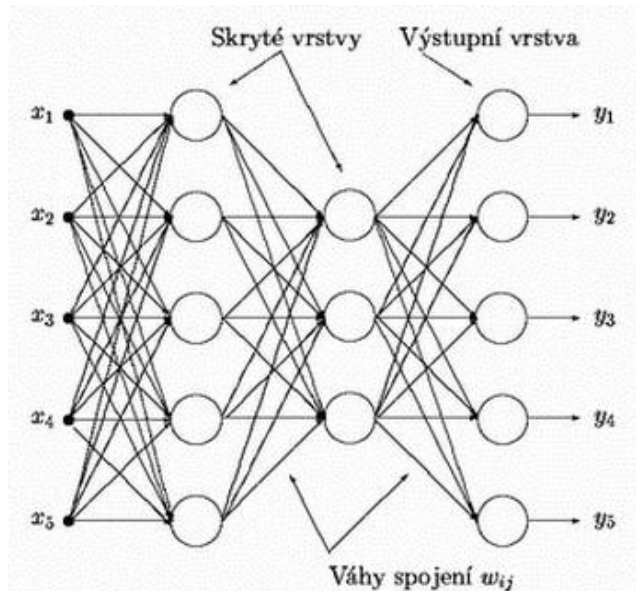


## 78 Vícevrstvý perceptron – struktura, způsoby učení, použití:

- Vícevrstvá síť s dopředným šířením (učení s učitelem)
- **Použití:** Klasifikace obrazů, Aproximace funkcí, Predikce časových řad, Řízení

### Trénování:

1. Počáteční inicializace vah a prahů jednotlivých neuronů
2. Přivedení vstupního vektoru a definice požadované výstupní odezvy
3. Vypočtení výstupních hodnot pro jednotlivé vrstvy sítě (dosazení sumy z  $w \cdot x$  – THÉTA do jednotlivých aktivačních funkcí)
4. Adaptace vah a prahů. Nastavení vah začíná u výstupních uzlů a rekurzivně postupuje ke vstupům.
5. Opakování kroků 3 a 4, dokud chyba není menší než předem stanovená hodnota.



### **79 Hopfieldova síť – struktura, modely, způsob učení, použití**

- Jednoduchá rekurentní síť s pevnými vahami
- Vstupní vektor musí být binární nebo bipolární (-1/1)
- **Použití:** Autoasociativní paměť, Rekonstrukce neúplných a šumem poškozených obrázků, Optimalizační problémy
- **Modely:** Asynchronní a Synchronní

### **80 Vyšší typy neuronových sítí – struktury, učení, použití**

ART:

- dvouvrstvá rekurentní síť (bez učitele)
- **použití:** shlukování, rozpoznávání znaků
- porovnávací a rozpoznávací vrstva

KOHONENOVA SÍŤ

- jednovrstvá s dopředným šířením (bez učitele)
- **použití:** shlukování, analýza dat, sémantické mapy
- porovnání mezi jednotkami – vítězný neuron zlepšuje svou pozici

### **81 Neocognitron – struktura, způsob učení, použití:**

- Vícevrstvá hierarchická neuronová síť
- **Použití:** rozpoznávání rukou psaných textů
- **Výhoda:** schopnost správně rozpoznávat nejen naučené obrazy, ale i obrazy, které vzniknou částečným posunutím, otočením nebo jinou deformací.

### **82 Základní úloha logiky**

Nalézání metod správného usuzování, tedy postupů, které dovolují přecházet od poznatků, jejichž pravdivost byla ověřena, k poznatkům novým z nich vyplývajících a tedy rovněž pravdivých.

### 83 Způsoby poznávání skutečnosti logickými systémy

- Přímé (empirické)
- Nepřímé (teoretické) – založeny na odvozování nových poznatků ze známých poznatků

### 84 Zaznamenávání zjištěných poznatků při identifikaci funkce logického systému:

Zjištěné poznatky jsou zaznamenány vhodným jazykem, u kterého je stejně důležitá syntax (symbol) jako sémantika (=význam).

Ze symbolů jazyka podle syntaktických pravidel můžeme vyjádřit **slova**. Správně sestavená slova nazýváme **formule**.

### 85 Co je „dedukce“ a „důkaz“:

- Dedukce – proces usuzování, kdy z předpokladů (premis) docházíme k závěru
- Důkaz – konečná posloupnost formulí vedoucí k závěru

### 86 Jakým způsobem dokazujeme platnost logických formulí:

- Formule tvořící výchozí soubor beru jako axiomy (nezvratitelné tvrzení)
- Pravdivost formulí dokážeme posloupností pravidel správného usuzování z tzv. premis

### 87 Vysvětlete pojmy „axiom“, „premise“, „konkluze“, „teorém“:

- Axiom – nezvratný výrok (tvrzení)
- Premisa - předpoklad, pravidlo správného usuzování
- Konkluze – to co vyplývá z výroku
- Teorém – formule, kterou dokazujeme ze souboru axiomů

### 88 Vysvětlete rozdíl mezi úplnou a neúplnou indukcí:

- Při úplné musíme projít všechny vzorky.
- Při neúplné se vezme jen část ze vstupních tvrzení a z toho se pak dokazuje.

*Příkladem neúplné indukce je například průzkum veřejného mínění. To se taky neotravují všichni lidi ale jen vzorek populace.*

### 89 Na příkladu vysvětlete pojmy „elementární výrok“ a „obecný výrok“

- Elementární výrok – ten který nejde dále rozdělit na jednodušší výroky „3 plus 4 je 7“
- Obecný výrok – je jakékoliv tvrzení, které může nabývat jen hodnot TRUE / FALSE „3 plus 4 je 7 a 3 plus 3 je 6“

### 90 Jaký je rozdíl mezi jednoduchým a složeným výrokem:

- Jednoduchý výrok = elementární výrok
- Složený výrok – pospojovaný z jednoduchých výroků, výrokových spojek a popřípadě závorek

### 91 Jaké pravdivostní hodnoty může nabýt výrok (výroková funkce)

Pravda (TRUE) nebo nepravda (FALSE)

## 92 Definice výrokové formule

Je to korektně vytvořený výraz pomocí symbolů výrokové logiky.

## 93 Vysvětlete pojmy „výroková tautologie“ a „výroková kontradikce“

- **Tautologie** – výrok je **pravdivý** bez ohledu na to jaké hodnoty jsou přiřazeny jednotlivým jeho členům
- **Kontradikce** – výrok je **nepravdivý** bez ohledu na to jaké hodnoty jsou přiřazeny jednotlivým jeho členům

## 94 Způsoby vyhodnocování logických výroků (jednoduchých i složených)

Pravdivostní tabulkou a pomocí De Morganových zákonů (pravidel)  $(A \cup B)' = A' \cap B'$

## 95 Charakterizujte logické vyplývání a uveďte jeho definici

Řekneme, že výroková formule B vyplývá z formule A, pokud je formule B pravdivá vždy, když je pravdivá formule A. A je tedy premisa a B je konkluze.

Zapisujeme  $A \models B$

## 96 Co je to „pravidlo odloučení“ (modus ponens)?

Jestliže „A“ a z „A plyne B“ platí, pak i „B“ platí.

## 97 Co říká „věta o dedukci“

Je-li věta B dokazatelná z vět  $A_1, A_2, \dots, A_n$ , pak věta  $A_n \models B$  je dokazatelná z vět  $A_1, A_2, \dots, A_{n-1}$

## 98 Definujte význam pojmů, že formule je „formálně dokazatelná“ a „logicky vyplývá“:

- **Formálně dokazatelná** – pokud lze formuli získat pomocí dedukce z axiomů
- **Logicky vyplývající** – B vyplývá z  $A_1, A_2, \dots, A_n$  jen tehdy když  $(A_1 \wedge A_2 \dots \wedge A_n) \Rightarrow B$  je výrokovou tautologií

## 99 Co patří do abecedy predikátové logiky 1. řádu:

- všechny symboly z abecedy výrokové logiky
- symboly pro popis objektů a jejich vlastností
- kvantifikátory  $\forall, \exists$

## 100 V predikátové logice 1. řádu vyjádřete obsah věty

„Někdo jezdí do práce autem, někdo veřejnou dopravou a někdo chodí pěšky“

$(\exists x \text{ JEZDIAUTEM}(x)) \wedge (\exists x \text{ JEZDIMHD}(x)) \wedge (\exists x \text{ CHODIPESKY}(x))$

## 101 Vysvětlete význam pojmů „term“, „atomická formule“, „formule“, „uzavřená formule“

- **Term** – konstanta nebo proměnná dosazovaná do funkce (formule)
- **Atomická formule** – výraz ve tvaru  $Dělitel(t_1, t_2)$ , kde Dělitel je m-místný predikátový symbol a tějka jsou termy
- **Formule** – spojení atomických formulí a symbolů pomocí symbolu z abecedy výrokové logiky (formule A, B spojené AND, OR,  $\Leftrightarrow, \Rightarrow$  )
- **Uzavřená formule** – formule kde se nevyskytují volné proměnné

## 102 Co je to klauzule? Jak je definována ?

= disjunkce literálů tzn.  $L1 \vee L2 \vee L3 \vee \dots$

Literál je výroková proměnná nebo její negace (také nazýváme atom)

## 103 Jak se vyhodnocují kvantifikované formule:

Odstraněním kvantifikátorů  $\forall, \exists$  (tj. částečným převedením do prenexní formy)

## 104 Vysvětlete pojmy, že formule predikátové logiky 1. řádu je „logicky pravdivá“, „splnitelná“, „nesplnitelná“, „není logicky pravdivá“ a sestavte pořadí platnosti:

- **Logicky pravdivá** = tautologie, pravdivá ve všech svých interpretacích
- **Logicky nepravdivá** - jestliže alespoň v jedné interpretaci nabývá hodnoty false
- **Splnitelná** – jestliže existuje alespoň jedna interpretace, kdy formule nabývá hodnoty true
- **Nesplnitelná** = kontradikce, jestliže ve všech interpretacích nabývá hodnoty false

## 105 Napište příklad disjunktivní a konjunktivní normální formy formule predikátové logiky 1. řádu

- Disjunktivní:  $(A \wedge B) \vee (A \wedge B \wedge C) \vee (B \wedge C)$
- Konjunktivní:  $(A \vee B) \wedge (A \vee B \vee C) \wedge (B \vee C)$

## 106 Vysvětlete podstatu rezoluční metody ve výrokové logice, uveďte vhodný příklad

Když není formule klauzulí, nahradí se ekvivalentem formulí, které jsou klauzulí.

## 107 Vysvětlete podstatu rezoluční metody v predikátové logice 1.řádu, uveďte příklad

Viz 106 – jen je potřeba volit správné substituce na volné proměnné ve formulích.

## 108 Vysvětlete pojmy „substituce“, „instance“, „základní instance“ a co je „unifikátor“

- **Substituce** – dosazení ekvivalentní formule za původní
- **Instance** – pokud provedu substituci tak výsledný tvar je instancí původního
- **Základní instance** – instance, která neobsahuje proměnné
- **Unifikátor** – když se literály po substituci rovnají

## 109 Jak se formálně definuje rezolventa klauzulí?

Jako disjunkce rodičovských klauzulí s vynecháním s vynecháním komplementárních literálů.

(komplementární je A a  $\neg(A)$ )

## 110 Co je „prenexní normální forma“ formule (PNF)

- pokud formule neobsahuje žádné kvantifikátory
- pomocí ekvivalentních úprav se dá každá formule převést do PNF

## 111 Převed'te formuli „ $\exists x P(x) \rightarrow \exists x Q(x)$ “ do prenexní normální formy

$\exists x P(x) \Rightarrow \exists x Q(x) \rightarrow \exists x (P(x) \Rightarrow Q(x)) \rightarrow \exists x (\neg P(x) \vee Q(x)) \rightarrow$

## 112 Vysvětlete pojem „rezolventa“. Jak ji získáme ?

Výsledek složení dvou rodičovských klauzulí obsahujících komplementární literály.

### 113 Rezoluční metodou dokažte platnost pravidla odvození (modus ponens)

### 114 Co je to „znalost“ a jaké má složky

Znalost je postup jak daná data interpretovat do souvislostí. Je považována za nejvyšší formu organizace strukturovaných dat.

**Složky:** prvky dat (respektive jejich reprezentanty), vlastnosti prvků, relace a operace nad prvky

### 115 Jak jsou vytvářeny znalosti a čím jsou tvořeny

Viz 114

### 116 Jaké druhy reprezentace znalostí rozlišujeme

- Deklarativní: konstatování skutečnosti (*Zanesený čistič vzduchu způsobí zvýšení spotřeby pohonných hmot.*)
- Procedurální: popis jak dosáhnout cíle (co máme dělat) (*Má-li automobil zvýšenou spotřebu pohonných hmot, ověř, zda nemá zanesený čistič vzduchu.*)

### 117 Formální systémy reprezentace znalostí:

- Logické systémy
- Produkční systémy – programy založené na pravidlech chování
- Rámce – OOP (vnitřní struktura uzlů)
- Sémantické sítě – realita jako objekty ve vztazích
- Procedurální systémy a speciální programovací jazyky

### 118 Uved'te příklad deklarativní reprezentace znalostí:

Zanesený čistič vzduchu způsobí zvýšení spotřeby pohonných hmot.

### 119 Uved'te příklad procedurální reprezentace znalostí:

Má-li automobil zvýšenou spotřebu pohonných hmot, ověř, zda nemá zanesený čistič vzduchu.

### 120 Vysvětlete pojmy „předmět reprezentace“, „prostředek reprezentace“ a „reprezentant“:

- Předmět reprezentace – to co chci reprezentovat
- Prostředek reprezentace – prostřednictvím čeho to chci reprezentovat
- Reprezentant – výrazový prostředek jak vyjádřit reprezentaci

### 121 Jaký typ datové struktury jsou reprezentované znalosti?

- tabulka, graf, strom, halda, spojový seznam

### 122 Jaké musejí být data a znalosti:

- přehledná, úplná, jednoznačná

### 123 Vysvětlete pojmy „fakta“ a „znalost“ – specifikujte rozdíly

**fakta** = to co víme × **znalost** = to co víme a používáme



Fakta jsou používána znalostmi k analýze toho, co se má provést – provedení operace, která je v datové struktuře znalosti

Báze znalostí obsahuje specifické znalosti z určité domény a specifické znalosti pro řešení problémů v této doméně.

Báze faktů se vytváří v průběhu řešení problému a obsahuje data ke konkrétnímu problému.

### 124 Odvozování nových faktů dopředným řetězením

- usuzování řízené daty – získáme data a na jejich základě rozhodujeme
- používá se při problémech zahrnujících syntézu
- *Př.: Co si dám k večeři. Zeptá se jakou mám chuť, jaké jsou zásoby v ledničce a ve spíži a podle toho doporučí recept.*

### 125 Odvozování nových faktů zpětným řetězením

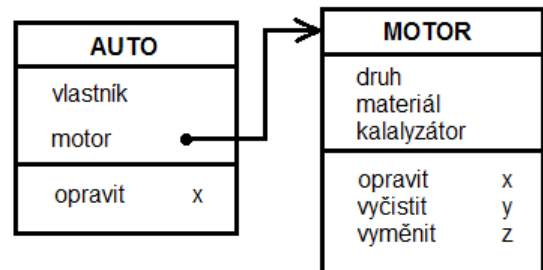
- usuzování řízené cíli
- expertní systém postupuje tak, že vybere nějaký cíl a pokouší se dokázat jeho platnost hledáním dat, která tento cíl podporují.
- tato strategie je vhodná pro diagnostické problémy s malým počtem cílových hypotéz
- *Př.: Dnes večer mám chuť na palačinky. Expertní systém mi na základě stavu zásob v ledničce a ve spíži řekne, zda je to možné.*

### 126 Uveďte příklad reprezentace znalostí logickým kalkulem, např. predikátovou logikou

A – motor se točí      B – žárovky svítí      C – jde nastartovat       $\neg A \wedge \neg B \rightarrow \neg C$

### 127 Uveďte příklad reprezentace znalostí rámci

- viz obrázek *Auto-Motor*

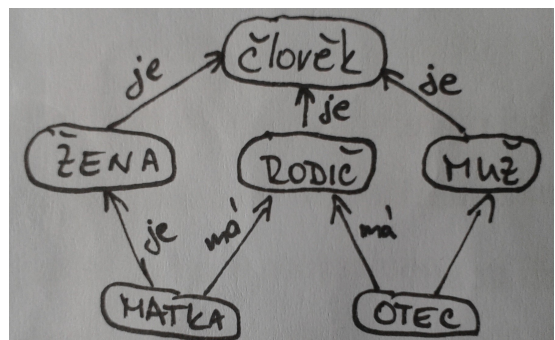


### 128 Uveďte příklad reprezentace znalostí sémantickou sítí

- viz obrázek *člověk-žena-rodíč-muž*

### 129 Uveďte příklad procedurální reprezentace znalostí

```
if (a > b) print a
else if (b > a) print b
else print „stejně“
```



### 130 Jak je definován rámec

Struktura dat pro reprezentaci stereotypních situací.

Ke každému rámci je přidružena informace.

V této informaci je obsaženo jak rámec používat, co je možné očekávat dále a co dělat pokud se očekávání nesplní.

### 131 Z čeho se skládá reprezentace uzlu rámcové sítě

Název uzlu, vlastnost uzlu (stopa), akce pokud se mění atribut stopy.

*Př.: Typ auta (název), barva (vlastnost), natankuj (akce)*

### 132 Z čeho se skládá reprezentace sémantické sítě

- uzly (objekty, koncepce, události,...)
- hrany (metody), které tvoří relace mezi uzly.  
Jako příklad hrany je nejčastější is-a (je) a has-part (má část)

### 133 K čemu přísluší pojem „stopa“

K rámcům. Jednotlivé atributy rámce se nazývají stopami.

### 134 Specifikujte rozdíl mezi „znalostním systémem“ a „expertním systémem“

Znalostní systém je obecnější.

Expertní systém vyžaduje už znalosti nějakého experta a často obsahuje také vysvětlovací podsystém.

### 135 Kdy je používán znalostní systém (na jaký typ úlohy)

Pro řešení takových úloh, které jsou všeobecně obtížné a jejichž uspokojivé řešení může provést pouze specialista (expert) v daném oboru.

Například: diagnostika závady, plánování, rozvrhování,...

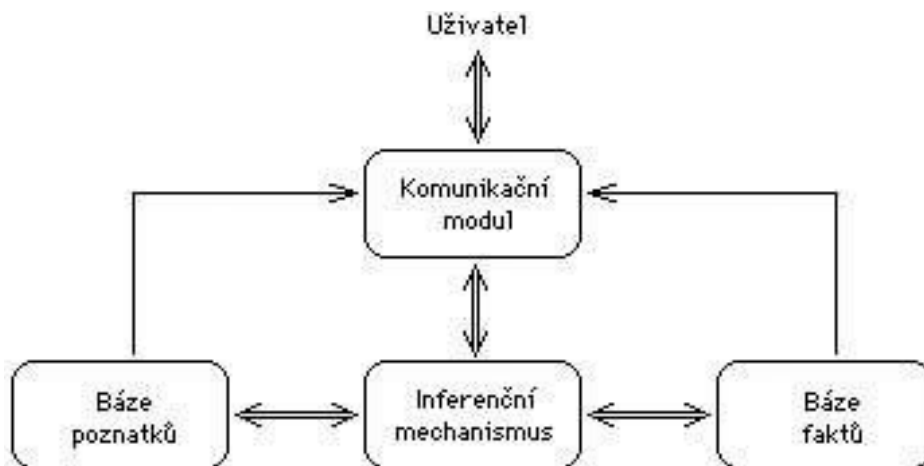
### 136 Jak rozdělujeme znalostní systémy

Z hlediska charakteru řešené úlohy rozlišujeme dva typy:

- Diagnostické – prochází data a snaží se určit, která z hypotéz nejlépe sedí na daný problém.
- Plánovací - pro úlohy, kdy je znám cíl řešení a počáteční stav a systém má s využitím dat o konkrétně řešeném případě nalézt (pokud možno optimální) posloupnost kroků, kterými lze cíle dosáhnout.

### 137 Nakreslete hrubou strukturu znalostního systému

138 Jaké jsou



**jednotlivé moduly znalostního systému:**

- **Báze faktů** – odpovídá množině dat klasického programu, většinou uloženy externě.
- **Báze poznatků** – struktury dat reprezentující znalosti odborníků, většinou uloženy ve vnitřní paměti počítače a během hledání mění bázi faktů
- **Inferenční mechanismus** – odvozovací algoritmus, který s využitím báze znalostí modifikuje bázi faktů, dokud nenalezne požadované řešení úlohy.
- **Komunikační model** – pro komunikaci s uživatelem

**139 K čemu znalostní systém používá vysvětlovací subsystém:**

Vysvětluje proč se znalostní systém (popřípadě expertní) rozhodl tak jak se rozhodl.

**140 Vysvětlete pojmy „inference“, „inferenční síť“ a „inferenční mechanismus“:**

- **Inference** = vyvozování logických závěrů z premis pokládaných za pravdivé
- **Inferenční síť** = graf, kde uzly jsou fakta a orientované hrany jsou pravidla.
- **Inferenční mechanismus** = mechanismus provádějící odvození

**141 Vysvětlete pojmy „data driven search“ a „goal driven search“**

- data driven search – dopředné zřetězení – od listů ke kořeni (aktualizace modelu inferenční sítě)
- goal driven search – zpětné zřetězení – od kořene (cíle) k listům (požadavkům na data)

**142 Co znamená zkratka NLP?**

Neuro-lingvistic programming (neurolingvistické programování)  
= **zpracování přirozeného jazyka**

**143 Jaké úlohy patří do oblasti zpracování přirozeného jazyka?**

- Vyhledávání textů
- Strojový překlad
- Podpora marketingu – třeba vyhledávání sentimentu v textu
- Podpora PR – třídění mailů
- Podpora rozpoznávání
- Oprava pravopisu
- Odpovídání na otázky

**144 Vyjmenujte typy systémů komunikujících přirozeným jazykem**

- Systémy typu „dotaz - odpověď“, Řečové portály
- Jednoduché konzultační systémy
- Dialogové informační systémy
- Znalostní a expertní systémy

**145 Co je „dialogový systém“?**

Systém, který dokáže komunikovat s uživatelem v jeho přirozeném jazyce.

**146 Co je třeba provést při analýze sdělení v přirozeném jazyce**

1. Snímání, reprezentace a předzpracování signálu
2. Segmentace a rozpoznání slov
3. Syntaktická analýza
4. Sémantická a programátorská analýza
5. Interpretace sdělení

**147 Uvedte jednotlivé kroky analýzy promluvy/věty přirozeného jazyka**

1. Signálová analýza
2. Akusticko-fonetická analýza
3. Lingvistická analýza
4. Extrakce významu a jeho interpretace

**148 Vysvětlete a popište úlohu rozpoznávání přirozeného jazyka**

1. zpracování akustického signálu – nahrávání, digitalizace, předzpracování, parametrizace
2. vytvoření akustického modelu  $P(O|W)$
3. vytvoření jazykového modelu  $P(W)$
4. nalezení nejčastější posloupnosti slov
  - patří k nejsložitějším aplikacím metod umělé inteligence
  - cílem je vytvořit programové a technické vybavení, které umožní výpočetnímu systému "porozumět" pokynům zadávaným člověkem běžnými větami přirozeného jazyka

**149 Vysvětlete pojem akustický model**

- jedná se o stochastický (opak deterministického) automat; generativní model
- obsahuje normální rozdělení
- jednotky: fóny, slabiky, 3-fóny

**150 Vysvětlete pojem jazykový model**

- určuje ppst posloupnosti N slov
- předpovídá další na základě předchozích

Například: korektura pravopisu, doplnění diakritiky

**151 Vysvětlete pojem N-gramový jazykový model**

- pravděpodobnostní přístup  $P(w) = P(w_1, \dots, w_n)$
- výpočet z relativní frekvence sekvencí slov na korpusu

Př.: Veronika je v kině. Jana je v zahradě.

Testovací sekvence: Veronika je v zahradě  $P(w) = ?$

$$P(„Veronika“) = P(„kině“) = P(„Jana“) = P(„zahradě“) = 1/8 = 0,125$$

$$P(„je“) = P(„v“) = 2/8 = 1/4 = 0,25 \quad \Rightarrow \quad P(w) = \left(\frac{1}{8}\right)^2 \cdot \left(\frac{1}{4}\right)^2$$

**152 Vysvětlete úlohu rozpoznávání pojmenovaných entit.**

- určení významu slov a slovních spojení, která mají předem daný význam
  - jména, města, státy, ...
- aby se nestal nesmyslný strojový překlad typu:
  - Justin Bieber ≠ just in beiber
  - Michal David – zpěvák nikoli dva lidé s křestními jmény Michal a David