

Obsah

1	Počítačová grafika a analýza obrazu	2
2	Matematické základy informatiky	3
3	Softwarové inženýrství	8
4	Databázové a informační systémy	9
5	Počítače a sítě	10

1 Počítačová grafika a analýza obrazu

- 1.1 Systémy barev v počítačové grafice, nelinearita grafického výstupu (gamma korekce), kompozice rastrových obrazů (alfa kanál), HDR.
- 1.2 Afinní a projektivní prostor. Afinní a projektivní transformace a jejich matematický zápis. Modelovací a zobrazovací transformace v počítačové grafice.
- 1.3 Metody získávání fotorealistických obrazů, rekurzivní sledování paprsku, radiometrie, zobrazovací rovnice, Monte Carlo přístupy ve výpočtu osvětlení, urychlovací metody.
- 1.4 Standardní zobrazovací řetězec a realizace jeho jednotlivých kroků, modely osvětlení a stínovací algoritmy, řešení viditelnosti, možnosti výpočtu globálního osvětlení v reálném čase, stručná charakteristika standardu OpenGL.
- 1.5 Komprese obrazu a videa, principy úprav obrazu v prostorové a frekvenční doméně.
- 1.6 Základní metody úpravy a segmentace obrazu (filtrace, prahování, hrany, oblasti, rohy).
- 1.7 Základní metody rozpoznávání objektů, příznakové rozpoznávání. Univerzální příznaky pro rozpoznávání (např. HOG), trénovací klasifikátory (např. SVM).
- 1.8 Hluboké neuronové sítě (např. konvoluční, popis jednotlivých vrstev).
- 1.9 Rekonstrukce 3D objektů z 2D obrazů (základní principy).

2 Matematické základy informatiky

2.1 Konečné automaty, regulární výrazy, uzávěrové vlastnosti třídy regulárních jazyků.

2.1.1 Konečný automat

- Výpočetní model, přijímá nebo zamítá slova reg. jazyka
- Znázornění tabulkou, grafem, stromem
- Deterministický
 - Definován uspořádanou pěticí
 - * Množina stavů
 - * Množina vstupních symbolů - abeceda
 - * Přechodová funkce
 - Vrací jeden stav
 - * Počáteční stav
 - * Množina koncových (přijímajících) stavů
- Nedeterministický
 - Definován uspořádanou pěticí
 - * Množina stavů
 - * Množina vstupních symbolů - abeceda
 - * Přechodová funkce
 - Vrací množinu stavů
 - * Množina počátečních stavů
 - * Množina koncových (přijímajících) stavů
 - Na rozdíl od deterministického
 - * Z jednoho stavu libovolný počet přechodů se stejným symbolem
 - * Nemusí ošetřovat všechny varianty
 - * Slovo přijímá, pokud existuje alespoň jeden výpočet vedoucí k přijetí
 - * Může mít více než jeden počáteční stav
 - * Lze ho převést na deterministický (až 2^n stavů v převedeném)
- Zobecněný nedeterministický
 - Nedeterministický + nulové ϵ (epsilon) přechody

2.1.2 Regulární výraz

- Řetězec, popisující celou množinu řetězců (regulární jazyk)
- Popisuje také konečný automat
- Znak abecedy, operace sjednocení, zřetězení a iterace.
- Obsahuje prázdné slovo ϵ a prázdný jazyk \emptyset

2.1.3 Uzávěrové vlastnosti třídy regulárních jazyků

- Uzavřenost – Výsledek operace s prvky množiny bude opět spadat do dané množiny
- Regulární výrazy uzavřené vůči
 - Sjednocení, průnik, doplněk
 - Zřetězení, iterace
 - Zrcadlový obraz

2.2 Bezkontextové gramatiky a jazyky. Zásobníkové automaty, jejich vztah k bezkontextovým gramatikám.

2.2.1 Bezkontextové gramatiky

- Definováno jako čtveřice
 - Množina neterminálů
 - * Neterminály = proměnné
 - Množina terminálů
 - * Terminály = Konstanty
 - Počáteční neterminál
 - Přepisovací pravidla
 - * Definují přepisování neterminálů
 - * Když není co přepisovat \rightarrow slovo, tvořené pouze neterminály
- Přijímáno zásobníkovým automatem
- Uzavřeno na sjednocení, zřetězení, iteraci a zrcadlový obraz
- Derivace slova – Konkrétní odvození slova z gramatiky
 - Levá – Přepisují se neterminály zleva
 - Pravá – Přepisují se neterminály zprava
- Derivační strom – Grafické znázornění derivace slova stromem
- Chomského normální forma
 - Neterminál \rightarrow neterminál ($A \rightarrow BC$)
 - Neterminál \rightarrow terminál ($A \rightarrow ab$)
 - Neterminál \rightarrow epsilon ($A \rightarrow \epsilon$)
- Nevypouštějící gramatika
 - Neobsahuje epsilon

2.2.2 Zásobníkový automat

- V podstatě nedeterministický konečný automat rozšířený o zásobník
- Díky zásobníku si pamatuje, kolik a jaké znaky přečetl
- Slovo přijme, pokud přečetl celé slovo a zásobník je prázdný

2.3 Matematické modely algoritmů -Turingovy stroje a stroje RAM. Složitost algoritmu, asymptotické odhady. Algoritmicky nerozhodnutelné problémy.

- Snaha popsat libovolný algoritmus \rightarrow Turingův a RAM stroj
- Teoretické modely univerzálních počítačů, programovacích jazyků

2.3.1 Turingův stroj

- Oboustranně nekonečná páska obsahující vstupní slovo
- Čtení i zápis pomocí hlavy
- Pohyb po pásce oběma směry
- Pracuje s celou abecedou
- Turingovská úplnost

- Stroj, počítač, programovací jazyk, úloha...
- Lze v něm odsimulovat libovolný jiný Turingův stroj
- Má stejnou výpočetní sílu jako Turingův stroj
- Church-Turingova teze
 - Ke každému algoritmu existuje ekvivalentní Turingův stroj

2.3.2 RAM stroj

- Model vycházející ze skutečných počítačů
- Slouží zejména k analýze algoritmů
 - Paměťová, časová složitost
- Skládá se z:
 - Programová jednotka
 - * Konečná posloupnost instrukcí \rightarrow program
 - Pracovní paměť
 - * Uspořádaná paměť
 - * Zápis celých čísel
 - * Indexace přirozenými čísly
 - $0 \rightarrow$ Pracovní registr
 - $1 \rightarrow$ Indexový registr
 - Vstupní a výstupní páska
 - * Sekvenční zápis/čtení celých čísel
 - Centrální jednotka
 - * Programový registr
 - * Ukazatel na instrukci
 - * Inkrementace ukazatele po provedení instrukce
 - $O 1$ nebo o více v případě skoku

2.3.3 Složitost algoritmů

- Pro srovnávání algoritmů řešících stejný problém
- Obecně, čím nižší, tím lepší
- Časová složitost
 - Jak závisí doba vykonávání na množství vstupních dat
- Prostorová složitost
 - Jak závisí potřebná paměť na množství vstupních dat
- Neudávají se konkrétní čísla, udává se funkce závislá na vstupních datech
 - Získáno simulací na RAM stroji
 - Počítání instrukcí pro nejhorší možný vstup
- Používá se asymptotická notace
 - $\mathcal{O}(f)$ – Horní ohraničení – Roste nejvýše tak rychle jako f
 - $\Theta(f)$ – Roste stejně rychle jako f
 - $\Omega(f)$ – Dolní ohraničení – Roste rychleji než f
 - ! Zanedbávají se konstanty
 - ! Analyzuje se nejhorší případ
 - * Relevantní může být spíše typický příklad

2.3.4 Rozhodnutelnost problémů

- Problém je rozhodnutelný, pokud existuje turingův stroj, který jej řeší
- Problém není rozhodnutelný, pokud pro daný vstup nenalezne žádný algoritmus výstup
- Ano/Ne problémy
 - Výstup z algoritmu může být Ano (true) nebo Ne (false)
 - Dají se na něj převést všechny problémy, pokud nepotřebujeme přesný výsledek
- Riceova věta
 - Nerozhodnutelná vlastnost, pokud je netriviální a vstupně/výstupní
 - Triviální vlastnost
 - * Mají ji všechny programy nebo ji nemá žádný program
 - * Triviální vlastnost je vždy vstupně/výstupní
 - Vstupně/výstupní vlastnost
 - * Pokud dva programy se stejnou IO tabulkou vlastnost mají nebo ne
- Částečná rozhodnutelnost
 - Pro očekávaný výstup ANO vrátí ANO
 - Pro očekávaný výstup NE vrátí NE nebo se program nezastaví

2.3.5 Algoritmicky nerozhodnutelné problémy

- Halting problem
- Ekvivalence bezkontextových gramatik
- Nejednoznačnost bezkontextové gramatiky

2.4 Třídy složitosti problémů. Třída PTIME a NPTIME, NP-úplné problémy.

2.4.1 Třída PTIME

- Rozhodovací (Ano/Ne) problémy
- Problémy, které lze řešit v polynomiálním čase
- Problémy v PTIME jsou prakticky zvládnutelné

2.4.2 Třída NPTIME

- Rozhodovací (Ano/Ne) problémy
- Problémy, které lze řešit v polynomiálním čase **nedeterministickými algoritmy**
 - Pro Ano stačí najít jedno řešení
 - Pro Ne je třeba důkaz, že žádné řešení nevrací Ano

2.4.3 NP-těžký problém

- Problém, na který lze převést každý problém z NPTIME v polynomiálním čase

2.4.4 NP-úplné problémy

- Problémy, patřící do NPTIME a zároveň jsou NP-těžké

- 2.5 Jazyk predikátové logiky prvního řádu. Práce s kvantifikátory a ekvivalentní transformace formulí.
- 2.6 Pojem relace, operace s relacemi, vlastnosti relací. Typy binárních relací. Relace ekvivalence a relace uspořádání.
- 2.7 Pojem operace a obecný pojem algebra. Algebry s jednou a dvěma binárními operacemi.
- 2.8 FCA – formální kontext, formální koncept, konceptuální svazy.
- 2.9 Asociační pravidla, hledání často se opakujících množin položek.
- 2.10 Metrické a topologické prostory – metriky a podobnosti.
- 2.11 Shlukování
- 2.12 Náhodná veličina. Základní typy náhodných veličin. Funkce určující rozdělení náhodných veličin.
- 2.13 Vybraná rozdělení diskrétní a spojitě náhodné veličiny - binomické, hypergeometrické, negativně binomické, Poissonovo, exponenciální, Weibullovo, normální rozdělení.
- 2.14 Popisná statistika. Číselné charakteristiky a vizualizace kategoriálních a kvantitativních proměnných.
- 2.15 Metody statistické indukce. Intervalové odhady. Princip testování hypotéz.

3 Softwarové inženýrství

- 3.1 Softwarový proces. Jeho definice, modely a úrovně vyspělosti.
- 3.2 Vymezení fáze „sběr a analýza požadavků“. Diagramy UML využité v dané fázi.
- 3.3 Vymezení fáze „Návrh“. Diagramy UML využité v dané fázi. Návrhové vzory – členění, popis a příklady.
- 3.4 Objektově orientované paradigma. Pojmy třída, objekt, rozhraní. Základní vlastnosti objektu a vztah ke třídě. Základní vztahy mezi třídami a rozhraními. Třídní vs. instanční vlastnosti.
- 3.5 Mapování UML diagramů na zdrojový kód.
- 3.6 Správa paměti(v jazycích C/C++, Java , C#, Python), virtuální stroj, podpora paralelního zpracování a vlákna.
- 3.7 Zpracování chyb v moderních programovacích jazycích, princip datových proudů – pro vstup a výstup. Rozdíl mezi znakově a bytově orientovanými datovými proudy.
- 3.8 Jazyk UML – typy diagramů a jejich využití v rámci vývoje.

4 Databázové a informační systémy

- 4.1 Modelování databázových systémů, konceptuální modelování, datová analýza, funkční analýza; nástroje a modely.
- 4.2 Relační datový model, SQL; funkční závislosti, dekompozice a normální formy.
- 4.3 Transakce, zotavení, log, ACID, operace COMMIT a ROLLBACK; problémy souběhu, řízení souběhu: zamykání, úroveň izolacev SQL.
- 4.4 Procedurální rozšíření SQL, PL/SQL, T-SQL, trigger, funkce, procedury, kurzory, hromadné operace.
- 4.5 Základní fyzická implementace databázových systémů: tabulky a indexy; plán vykonávání dotazů.
- 4.6 Objektově-relační datový model a XML datový model: principy, dotazovací jazyky.
- 4.7 Datová vrstva informačního systému; existující API, rámce a implementace, bezpečnost; objektově-relační mapování.
- 4.8 Distribuované SŘBD, fragmentace a replikace.

5 Počítače a sítě

- 5.1 Architektura univerzálních procesorů. Principy urychlování činnosti procesorů.
- 5.2 Základní vlastnosti monolitických počítačů a jejich typické integrované periférie. Možnosti použití.
- 5.3 Protokolová rodina TCP/IP.
- 5.4 Metody sdíleného přístupu ke společnému kanálu.
- 5.5 Problémy směrování v počítačových sítích. Adresování v IP, překlad adres (NAT).
- 5.6 Bezpečnost počítačových sítí s TCP/IP: útoky, paketové filtry, stavový firewall. Šifrování a autentizace, virtuální privátní sítě.