

# Programové a informačné systémy

## 2-INF-144 Kompilátory

- Štruktúra a použitie kompilátorov, nástroje pre tvorbu kompilátorov.
  - Základné pojmy a štruktúra kompilátora (lexikálna analýza, syntaktická analýza, ..., rozdelenie na front-end a back-end).
  - Vzťah k teórii jazykov. Syntax vs. sémantika.
  - Kompilácia do iných jazykov, byte-kódu, strojového kódu. Porovnanie kompilátorov s interpretermi, JIT.
  - Využitie kompilátorov pri statickej kontrole, formátovaní textu, spracovaní štruktúrovaných dokumentov.
  - Nástroje pre tvorbu kompilátorov (Flex, Bison alebo ANTLR)
  - Jednoduchý jednoprechodový kompilátor, rekurzívny zostup.
- Lexikálna analýza.
  - Dôvody oddelenia lexikálnej a syntaktickej analýzy. Token, Vzor, Lexéma.
  - Lexikálna analýza a odsadenie blokov ako syntaktická konštrukcia.
  - Regulárne jazyky, Nedeterministické konečné automaty, deterministické konečné automaty (DKA) a prechod od regulárnych jazykov k DKA.
  - Vyhľadávanie v texte (napr. Aho-Corasick).
  - Oprava chýb a zotavenie sa z chýb pri lexikálnej analýze.
- Syntaktická analýza zhora-nadol.
  - Bezkontextové gramatiky. Strom ododenia. Abstraktný syntaktický strom.
  - Úprava gramatiky (eliminácia nejednoznačnosti a ľavej rekurzie, ľavá faktorizácia).
  - Rekurzívny zostup, LL(1), LL(k).
  - Všeobecné metódy pre bezkontextové jazyky: Earley-ho algoritmus, Cocke-YoungerKasami.
  - Oprava chýb a zotavenie sa z chýb pri syntaktickej analýze.
- Syntaktická analýza zdola-nahor.
  - Bezkontextové gramatiky. Strom ododenia. Abstraktný syntaktický strom.
  - Úprava gramatiky – eliminácia nejednoznačnosti.
  - Operátorovo precedenčné parsovanie.
  - LR parsovanie, model LR parsera, LR parsovací algoritmus.
  - LR metódy – SLR, CLR a LALR.
  - Oprava chýb a zotavenie sa z chýb pri syntaktickej analýze.
- Syntaxou riadené preklady.
  - Atribútové gramatiky. Syntaxou riadená definícia.
  - Syntetizované a zdedené atribúty. S a L-atribútová definícia.
  - Vyhodnocovanie atribútov. Graf závislosti atribútov. Test na cirkularitu.
  - Porovnanie metód zhora-nadol a zdola-nahor.
- Podpora v čase behu (Run-time environments).
  - Organizácia a pridelovanie pamäte (statické dáta, zásobník, dynamická pamäť). Zber smetí (garbage collection).
  - Volanie funkcií a procedúr, dovzdávanie parametrov, lokálne a nelokálne premenné, aktivačný záznam, display.
  - Rozsah platnosti (scope) – statický (lexikálny) vs. dynamický.

- Generovanie medzijazyka.
  - Formy medzijazyka: Poľská bezzátvorková forma, štvorice (trojadresové inštrukcie) a trojice.
  - Generovanie medzijazyka syntaxou riadeným prekladom (atribúty kód a umiestnenie).
  - Popis základných konštrukcií: aritmetické a logické výrazy, príkaz priradenia, riadiace konštrukcie (if, while,...), volanie procedúr a funkcií.
  - Spätné plátanie (backpatching).
- Generovanie kódu.
  - Cieľový počítač: CISC, RISC. Dĺžka a ceny inštrukcií. Zohľadňovanie cache a pipeline pri preklade. Špecifiká cieľových inštrukčných sád, ktoré treba zohľadňovať.
  - Pridelovanie registrov. Pridelovanie registrov farbením grafu (register inference graph).
  - Administrácia pamäte. Statická pamäť, zásobník, dynamická pamäť – heap.
  - Rozdelenie programu na bloky, generovanie kódu z dagu.
  - Generovanie kódu pokrývaním ozdobeného stromu.
- Optimalizácia kódu.
  - Optimalizácia základných blokov, optimalizačné transformácie.
  - Eliminácia mŕtveho kódu, kukátková (peephole) optimalizácia, algebraické identity, redukcia v sile, vyhodnocovanie konštantných podvýrazov.
  - Analýza toku dát. Riešenie rovníc toku dát.

## 2-INF-184 Programovacie jazyky

- Zoznamy, n-tice, algebraické dátové typy (ADT), zovšeobecnené algebraické dátové typy (GADT), abstraktné dátové typy a moduly.
- Pattern matching a kontrola jeho úplnosti, chvostová rekúzia, akumulátor, typy n-tého ranku (Rank-N types).
- Striktné (strict), odložené (delayed) a lenivé (lazy) vyhodnocovanie parametrov. Odloženie vyhodnotenia parametrov v striktných jazykoch (thunks). Vynútenie vyhodnotenia parametrov v lenivých jazykoch (foldl vs. foldl'). Meniteľné (mutable) a nemenné (immutable) dátové štruktúry.
- Návrhový vzor monad a jeho využitie (Maybe, List, Reader, Writer, State).
- Prvorádové (first-class) funkcie, funkcie vyššieho rádu (map, foldr, foldl), uzávery (closures), typová inferencia.
- Funkcie viacerých argumentov, n-tice, currying vs. čiastočné aplikovanie funkcie, bezbodový (pointfree) štýl zápisu funkcie.
- Staticky a dynamicky typované jazyky, jazyky bez typovej kontroly, možné rozdiely v sémantike príkazu if a zoznamov, duck typing.
- Funkcionálna vs. OOP dekompozícia, mixins, double dispatch, subtyping, generics, polymorfické funkcie.
- LISP syntax, rozdiel medzi reťazcami a symbolmi, eval, quasiquote, unquote, string interpolation.
- Makrá, kedy je ich vhodné použiť, hygienické makrá.

## 2-INF-145 Tvorba internetových aplikácií

- Základy internetových aplikácií.
  - HTTP, HTML, CSS, SVG, JavaScript.
  - Oddelenie prezentácie, obsahu a kódu, šablóny.
  - JavaScript transpilers.
  - jQuery – výhody a nevýhody.
  - Prechod od statických stránok k dynamickým aplikáciám (cookies, sessions, upload, dynamické generovanie obsahu – databáza).
- JavaScript OOP.
  - Objektovo orientované programovanie založené na prototypoch (rozdiely oproti OOP založenému na triedach).
  - Rozdiel medzi \_\_proto\_\_ a prototype položkami objektov.
  - Manipulácia s this pomocou call, apply a bind a jej využitie.
  - Privátne položky a metódy v JavaScript objektoch a ich súvis s closures.
- JavaScript.
  - Typované polia (buffer, view, DataView).
  - Strict mode.
  - Funkcie s premenlivým počtom argumentov.
  - Block vs. function scoping (t.j. let vs. var).
  - Reťazenie (kaskádovanie) metód.
- Canvas element a Scalable Vector Graphics (SVG).
  - Porovnanie <canvas> elementu a SVG (nezávislosť na rozlíšení, DOM, CSS, udalosti, odkazy, ...).
  - Použitie <g> elementu (zoskupovanie), <use> elementu (znovupoužitie) a <def> elementu (definície bez zobrazenia) pri SVG.
  - Transformácia súradnicového systému, gradienty, zarovnanie textu na cestu.
  - Data URL format a <img> element.
- AJAX a WebSockets.
  - Princíp AJAX, XMLHttpRequest, výhody a nevýhody.
  - Notifikácie od servera pred WebSocketmi (Polling, Long Polling, Streaming).
  - Výhody WebSocketov (šetrí šírku pásma, CPU, jednoduché API, súčasť HTML5).
- Bezpečnosť internetových aplikácií
  - HTTPS, identifikácia a autentifikácia používateľov.
  - Session hijacking, session riding.
  - URL tampering, forceful browsing.
  - HTML injection, SQL injection.
  - White-list vs. black-list.

## 2-INF-183 Počítačové siete 2

- Riadenie prístupu k zdieľanému médiu v počítačových sieťach, predpoklady a vlastnosti
  - Ethernet (CSMA/CD)
  - WiFi (CSMA/CA)
  - DOCSIS
- Prenosy paketov cez jednoduché sériové linky
  - kódovanie signálov
  - synchronizácia hodín
  - PPP

- Spanning Tree protokoly
  - STP
  - RSTP
  - MSTP
- Smerovacie protokoly v TCP/IP
  - OSPF
  - BGP
- pôsobý multiplexovania v sieťach
  - TDMA
  - FDMA

## 2-INF-222 Objektová analýza a modelovanie

- Popis softverovo intenzívneho systému
  - Aké aspekty softverovo intenzívneho systému je potrebné zachytiť pre vytvorenie uceleného pohľadu na systém?
  - Použitie jednotlivých UML diagramov na zachytenie vlastností softveru.
- Fázy vývoja softveru
  - Popíšte najbežnejšie metodológie vývoja softveru.
  - Použitie UML diagramov v jednotlivých fázach vývoja softveru
- Požiadavky, analýza požiadaviek, use case diagramy
  - zdroje požiadaviek
  - metódy zachytávania požiadaviek
  - analýza požiadaviek
  - typy požiadaviek
  - organizácia požiadaviek v rámci dokumentácie softverového projektu
  - základné elementy use-case diagramov
- Analýza a plánovanie v iteratívno-inkrementálnom vývoji
  - typický obsah prvých iterácií softverového projektu
  - ako sa mení typický rozsah iterácie počas vývoja softvéru
  - ktorým položkám je potrebné pri analýze venovať zvýšenú pozornosť a ako ich spoznáme
  - ako sa určuje rozsah iterácie (porovnať rôzne prístupy, jednak z obsahovej a aj z časovej/množstevnej stránky)
- Class diagramy, designové princípy
  - základné elementy class a object diagramov
  - použitie object a class diagramov na popis softvéru
  - základné princípy (nielen) objektovo-orientovaného dizajnu
- Základné návrhové vzory
  - základné vytvárajúce vzory, štrukturálne vzory a vzory správania
  - pomocou UML opíšte vybrané návrhové vzory
- Immutable datové štruktúry, testovateľnosť OO kódu.
  - Interface immutable tried, implementácia immutable verzii základných štruktúr, výhody a nevýhody immutable datových štruktúr.
  - Aké prvky v návrhu a implementácii tried sťažujú testovanie? Ako navrhnuť objekt tak, aby sa dal jednoduchšie testovať?
  - Ako sa pri unit testoch vysporiadať so závislosťami triedy

- Activity diagramy, sequence diagramy, communications diagramy a state machines
  - základné elementy diagramov.
  - využitie diagramov popis softveru.
  - vysvetlite fungovanie vybraného návrhového vzoru, pričom vo vašom vysvetlení má byť kľúčový activity diagram alebo state machine.
- Ostatné UML diagramy
  - Elementy a použitie komponent diagramov, composite structure diagramov, package diagramov, deployment diagramov
  - UML profily
- Objekty a databázy
  - Nekompatibilitnosť OO systémov a relačných databáz
  - ORM
  - No-SQL databázy

## 2-INF-500 Databázy

- Sémantika dotazov.
  - Teórie, modely, stratifikovaná negácia, lokálne stratifikovaná negácia.
  - Sémantika pevného bodu.
  - Gelfond-Lifschitzova transformácia, stabilné modely, well-founded model.
- Optimalizácia rekurzívnych dotazov.
  - Datalogové programy s funkčnými symbolmi, unifikácia termov, naivná evaluácia, seminaivná evaluácia.
  - Prolog, SLD rezolúcia, rule-goal-tree, rule-goal-graph, väzby medzi premennými (binding patterns).
  - Magická transformácia, zovšeobecnená magická transformácia.
- Optimalizácia nerekurzívnych dotazov.
  - Strom relačného výrazu, hypergraf dotazu.
  - Dekompozícia hypergrafu, Wong-Youssefiho algoritmus.
  - Semijoin, úplný reduktor, Yannakakisov algoritmus.
  - Optimalizácia poradia joinov.
  - Odhad veľkosti výstupu dotazu.
  - Použitie indexov, dekompozícia databázy.
  - Pohltlenie (containment) a ekvivalencia konjunktívnych dotazov.
  - Testovanie pohltlenia dotazov, petrifikované dotazy, kanonické databázy.
  - Tablá, vynútenie závislostí, slabá ekvivalencia dotazov, minimalizácia tabiel.
- Vyššie normálne formy.
  - Funkčné závislosti, 3NF, BCNF.
  - Multizávislosti, 4NF.
  - Joinovacie závislosti, 5NF.
  - Inklúzne závislosti.
  - DKNF (domain-key normal form).
  - Metodika normalizácie.
- Distribuované databázy.
  - Model distribuovanej databázy.
  - Atomický commit. Požiadavky na ACP.
  - Dvojfázový ACP. Trojfázový ACP. Výber koordinátora, bully algoritmus.
  - Replikácia dát, distribuované zamykanie, distribuované deadlocky. Quorum protokol.
  - Synchronizácia fyzických hodín, Christianov protokol, Berkeley protokol.
  - Časové pečiatky, Lamportov protokol.