

UNIVERZITA HRADEC KRÁLOVÉ



## Státní závěrečné zkoušky

Aplikovaná informatika - 3

# Obsah

<b>1</b>	<b>PROG</b>	<b>1</b>
1.1	Logický program - struktura, základní pojmy, datová struktura seznam, práce s databází Prologu. Hlavní odlišnosti oproti procedurálnímu programování, možnosti použití neprocedurálního programovacího jazyka.	2
1.2	Databáze, databázový systém. Hlavní funkce DBS. Historický vývoj DBS. Modely dat. Relační algebra: projekce, selekce, spojení. SQL. . . . .	2
1.3	Konceptuální modelování. E-R model a jeho grafické znázornění. Relační model. Typy vztahů mezi entitami a jejich reprezentace v relačním modelu. Vlastnosti relační tabulky. Normální formy relačního schématu.	2
1.4	Ontologické inženýrství: pojem ontologie v kontextu informatiky, základní stavební prvky ontologií, typy ontologií, jazyky ontologického modelování, návrhové vzory, normalizace ontologie. Odvozování nad ontologií (kontrola konzistence, klasifikace), nástroje, použití ontologií. .	2
1.5	Sémantický web: technologie sémantického webu, metadata, RDF, RDFS, OWL, dotazování se na sémantický web (význam, jazyky), sémantický web a odvozování (význam, jazyky), aplikace sémantického webu. . . .	2
1.6	Námětové mapy: standard Topic Maps a jeho součásti, základní stavební prvky námětové mapy, postup tvorby námětové mapy, implementace námětových map (prostředí, syntaxe), dotazování se na námětové mapy, odvozování s námětovými mapami, aplikace námětových map. . . . .	2
1.7	Objektové modelování a programování - základní pojmy, podstata, využití. Softwarový proces. UML. Událostmi řízené programování. Architektura MVC. . . . .	3
1.7.1	Modelování . . . . .	3
1.7.2	Událostmi řízené programování - Event-driven programming . .	3
1.7.3	Architektura MVC . . . . .	4
1.8	Práce s kolekcemi – typy kolekcí, příklady použití, algoritmy pracující nad kolekcemi (řazení, vyhledávání), základní principy implementace ve zvoleném programovacím jazyce. . . . .	5
1.9	Problematika perzistentního (trvalého) ukládání dat ve vybraném programovacím jazyce. . . . .	5

1.10	Webové aplikace – principy, nástroje. Vícevrstvé aplikace. Zabezpečení aplikace. . . . .	5
1.11	Základní algoritmy a principy počítačové grafiky – metody vizualizace, určení viditelnosti a osvětlení, reprezentace grafické informace, OpenGL. . . . .	5
1.12	Základy zpracování obrazu a počítačového rozpoznávání – metody snímání, předzpracování, segmentace a klasifikace obrazu, formáty pro ukládání rastrového obrazu, komprese, barva a barevné modely. . . . .	5
1.13	Algoritmy pracující s grafy. Prohledávání grafů do hloubky a do šířky, využití prohledávání grafů v dalších úlohách. . . . .	5
<b>2</b>	<b>TECH</b>	<b>6</b>
2.1	Principy počítačů (historický vývoj, předpoklady fungování, binární logika, modulace signálu). . . . .	7
2.2	2. Architektura počítače (von Neumannovo a Harwardské schéma, Flynnova taxonomie, základní deska, procesor, mikroarchitektura procesoru, paměti, sběrnice, řadič, přídavné karty, ovladače). . . . .	7
2.3	Paměťový systém počítače a ukládání dat (typy, principy fungování, frekvence, normy, logická a fyzická struktura disku, RAM, ROM, Cache, HDD, CD, DVD, FLASH...) . . . . .	7
2.4	Architektura periferních zařízení (rozdělení, principy, funkce, typy, rozhraní, příklady) . . . . .	7
2.5	Servery a pracovní stanice (rozdíly, kritéria výběru, role serverů, serverové technologie, zálohování dat včetně RAID) . . . . .	7
2.6	Komunikační prostředky (principy komunikace, modulace signálu, rozdělení a porovnání, média, mobilní technologie) . . . . .	7
2.7	ETHERNET (principy fungování, vývoj a topologie, přístupová metoda, síťová karta, strukturovaná kabeláž) . . . . .	7
2.8	RM ISO/OSI, TCP/IP (popis a srovnání, funkce zásadních protokolů, IP adresy) . . . . .	7
2.9	Internet (organizační struktura, vývoj, RFC dokumenty, domény, technické předpoklady pro připojení, hrozby) . . . . .	7
2.10	Směrování (základní principy, směrovací protokoly, směrovací algoritmy, směrovače) . . . . .	7
2.11	Propojování a management sítí (přenosová média, technologie pro různé vrstvy, WIFI, VPN, systémy pro vzdálený přístup, řešení založená na SNMP) . . . . .	7

---

2.12 Principy operačních systémů (základní rozdělení, druhy operačních systémů, procesy, správa procesů a systémových zdrojů, uživatelská rozhraní). . . . .	7
2.13 Souborové systémy a logická struktura dat (principy, porovnání, příklady). . . . .	7
2.14 Operační systémy Windows (principy MS DOS, MS Windows, architektura, verze, funkce, rozdíly). . . . .	7
2.15 Operační systémy Unix, Linux, BSD, MacOS (základní myšlenky, výhody a nevýhody, open-source, vznik a vývoj, licence, distribuce, základy ovládání - shell, rozdíly, historie a vývoj) . . . . .	7
2.16 Serverové operační systémy (specifika serverových operačních systémů, rozdíly mezi OS pro osobní počítač a pro server, serverové služby, správa uživatelů) . . . . .	7
<b>Seznam zdrojů</b>	<b>8</b>



# **1 PROG**

- 1.1 Logický program - struktura, základní pojmy, datová struktura seznam, práce s databází Prologu. Hlavní odlišnosti oproti procedurálnímu programování, možnosti použití neprocedurálního programovacího jazyka.**
- 1.2 Databáze, databázový systém. Hlavní funkce DBS. Historický vývoj DBS. Modely dat. Relační algebra: projekce, selekce, spojení. SQL.**
- 1.3 Konceptuální modelování. E-R model a jeho grafické znázornění. Relační model. Typy vztahů mezi entitami a jejich reprezentace v relačním modelu. Vlastnosti relační tabulky. Normální formy relačního schématu.**
- 1.4 Ontologické inženýrství: pojem ontologie v kontextu informatiky, základní stavební prvky ontologií, typy ontologií, jazyky ontologického modelování, návrhové vzory, normalizace ontologie. Odvozování nad ontologií (kontrola konzistence, klasifikace), nástroje, použití ontologií.**
- 1.5 Sémantický web: technologie sémantického webu, metadata, RDF, RDFS, OWL, dotazování se na sémantický web (význam, jazyky), sémantický web a odvozování (význam, jazyky), aplikace sémantického webu.**
- 1.6 Námětové mapy: standard<sup>2</sup> Topic Maps a jeho součásti, základní stavební prvky námětové mapy, postup tvorby námětové mapy, implementace námětových map (prostředí, syntaxe), dotazování se na námětové mapy, odvozování s námětovými mapami, aplikace**

## **1.7 Objektové modelování a programování - základní pojmy, podstata, využití. Softwarový proces. UML. Událostmi řízené programování. Architektura MVC.**

### **1.7.1 Modelování**

vychází z faktu, že budovaný IS je obrazem (modelem) vnějšího světa odráží tedy realitu a struktura systému vychází z reality nelze namodelovat přesně celou realitu => abstrakce – odstranění nepodstatných prvků lze takto modelovat i velmi rozsáhlé systémy modelování umožňuje usnadnění komunikace se zákazníkem lepší přehled o aktuálním stavu projektu snazší vytváření dokumentace atd. hlavní principy modelování

**abstrakce**

**formalizace**

**jednoznačnost**

to, co pro mě není důležité, odstíním je přesně dané, co se jak vyjadřuje usnadnění komunikace v týmu i se zákazníkem vyplývá z formalizace; každý prvek lze jednoznačně identifikovat a popsat snížení nadbytečnosti princip tří architektur postupná tvorba tří typů architektur každá má jinou míru abstrakce, logiku a hloubku popisu, ale každá úplně popisuje celý systém jde o konceptuální úroveň – popisuje obsah, ne formu (CO?) technologická úroveň – popisuje technologii, která bude použita (JAK?) fyzická úroveň – popisuje detaily implementace (ČÍM?)

### **1.7.2 Událostmi řízené programování - Event-driven programming**

Událost (Event) vzniká buď jako výsledek interakce GUI s uživatelem nebo jako důsledek změny vnitřního stavu aplikace či OS Obsluhou události nazýváme úsek kódu, který je při vzniku události automaticky vyvolán a provádí činnost k události připojenou (někdy také ohlasová metoda události či Event Handler). Příklady typů událostí:

- Klik/DvojKlik
- Zaměření/Ztráta zaměření
- Změna stavu komponenty
- Stisk, uvolnění klávesy
- Stisknutí, uvolnění tl. myši
- Posun myši

- Událost časovače
- Zpráva systému

### 1.7.3 Architektura MVC

(model-view-controller) – zřejmě odpovídá úrovní modelování tříd (viz výše). Lepší by bylo psát VCM nebo MCV, neboť to pak odpovídá vztahům v architektuře. rozděluje datový model aplikace, uživatelské rozhraní a řídicí logiku do tří nezávislých komponent že modifikace některé z nich má minimální vliv na ostatní. vytváření aplikací s využitím architektury MVC vyžaduje vytvoření tří komponent Model (model), což je doménově specifická reprezentace informací, s nimiž aplikace pracuje. View (pohled), který převádí data reprezentovaná modelem do podoby vhodné k interaktivní prezentaci uživateli. Controller (řadič), který reaguje na události (typicky pocházející od uživatele) a zajišťuje změny v modelu nebo v pohled. Příklad: ve skriptovacím jazyku pro WWW Ruby on Rails dělají: Model - z databáze vycucne řádky z tabulky odpovídající danému modelu a převede je na objekt (sloupce tabulky= vlastnosti objektu). Nebo kolekci objektů. Controller - z Pohledu získá informaci, co chce uživatel vidět (provést), zpracuje požadavek, od modelu si vyžádá patřičné objekty, ty zpracuje (business logic) a předá je patřičnému pohledu (odpovídá požadované akci, nemusí to být tentýž co to celé vyvolal) View - vezme zpracované objekty od controleru a vhodně je zobrazí (nijak je neupravuje, to je práce controleru, který volá metodu v modelu) a čeká co si bude páníček přát dále.



- 1.8 Práce s kolekcemi – typy kolekcí, příklady použití, algoritmy pracující nad kolekcemi (řazení, vyhledávání), základní principy implementace ve zvoleném programovacím jazyce.**
- 1.9 Problematika perzistentního (trvalého) ukládání dat ve vybraném programovacím jazyce.**
- 1.10 Webové aplikace – principy, nástroje. Vícevrstvé aplikace. Zabezpečení aplikace.**
- 1.11 Základní algoritmy a principy počítačové grafiky – metody vizualizace, určení viditelnosti a osvětlení, reprezentace grafické informace, OPENGL.**
- 1.12 Základy zpracování obrazu a počítačového rozpoznávání – metody snímání, předzpracování, segmentace a klasifikace obrazu, formáty pro ukládání rastrového obrazu, komprese, barva a barevné modely.**
- 1.13 Algoritmy pracující s grafy. Prohledávání grafů do hloubky a do šířky, využití prohledávání grafů v dalších úlohách.**



## **2 TECH**

- 2.1 Principy počítačů (historický vývoj, předpoklady fungování, binární logika, modulace signálu).**
- 2.2 2. Architektura počítače (von Neumannovo a Harwardské schéma, Flynnova taxonomie, základní deska, procesor, mikroarchitektura procesoru, paměti, sběrnice, řadič, přídatné karty, ovladače).**
- 2.3 Paměťový systém počítače a ukládání dat (typy, principy fungování, frekvence, normy, logická a fyzická struktura disku, RAM, ROM, Cache, HDD, CD, DVD, FLASH...)**
- 2.4 Architektura periferních zařízení (rozdělení, principy, funkce, typy, rozhraní, příklady)**
- 2.5 Servery a pracovní stanice (rozdíly, kritéria výběru, role serverů, serverové technologie, zálohování dat včetně RAID)**
- 2.6 Komunikační prostředky (principy komunikace, modulace signálu, rozdělení a porovnání, média, mobilní technologie)**
- 2.7 ETHERNET (principy fungování, vývoj a topologie, přístupová metoda, síťová karta, strukturovaná kabeláž)**
- 2.8 RM ISO/OSI, TCP/IP (popis a srovnání, funkce zásadních protokolů, IP adresy)**
- 2.9 Internet (organizační struktura, vývoj, RFC dokumenty, domény, technické předpoklady pro připojení, hrozby)**
- 2.10 Směrování (základní principy, směrovací protokoly, směrovací algoritmy, směrovače)**

třeby správy sítě, a jejich následné vyhodnocování. Na tomto protokolu je dnes založena většina prostředků a nástrojů pro správu sítě.

Má tři verze: druhá obsahuje navíc autentizaci a třetí šifrování. Nejvíce zařízení podporuje druhou verzi.

Rozlišuje se mezi stranou monitorovanou (hlídaný systém) a monitorovací (sběrna dat). Tyto strany mohou běžet buď odděleně na různých fyzických strojích, nebo v rámci jednoho stroje. Na monitorované straně je spuštěn agent a na straně monitorovací manager. Na straně monitorované jsou operativně shromažďovány informace o stavu systému (zařízení). Manager vznáší požadavky agentovi, zpravidla na zaslání požadovaných informací (zpráv). Agent zajišťuje realizaci reakcí na požadavky managera. Získaný obsah zpráv se na straně monitorovací může dále různým způsobem zpracovávat (tabulky, grafy, ...). Komunikace mezi agentem a managerem se označuje jako SNMP operace.

Na straně monitorované může existovat možnost takové konfigurace, kdy agent zašle managerovi informace (SNMP MESSAGE) automaticky bez jeho požadavku. K tomu dojde zpravidla potom, kdy byla splněna předem definovaná podmínka (výpadek, kolize, dosažení hraniční hodnoty, ...), agent nečeká na odpověď. Takové konfiguraci agenta se říká SNMP TRAP (tzv. past na události).

- 2.12 Principy operačních systémů (základní rozdělení, druhy operačních systémů, procesy, správa procesů a systémových zdrojů, uživatelská rozhraní).**
- 2.13 Souborové systémy a logická struktura dat (principy, porovnání, příklady).**
- 2.14 Operační systémy Windows (principy MS DOS, MS Windows, architektura, verze, funkce, rozdíly).**
- 2.15 Operační systémy Unix, Linux, BSD, MacOS (základní myšlenky, výhody a nevýhody, open-source, vznik a vývoj, licence, distribuce, základy ovládání - shell, rozdíly, historie a vývoj)**
- 2.16 Serverové operační systémy (specifika serverových operačních systémů, rozdíly mezi OS pro osobní počítač a pro server, serverové služby, správa uživatelů)**

# Literatura