



Univerzita Palackého
v Olomouci

Genius loci...

Okruhy pro ústní část státní závěrečné zkoušky v akademickém roce 2024/2025

studijní program: B0613A140019 Informatika
specializace: Programování a vývoj software
typ: bakalářský
forma: prezenční

Pro ústní zkoušku se stanovují následující okruhy. Student si vylosuje tři otázky, které do okruhů tematicky spadají (pro každý okruh jednu otázku). Zkouška trvá cca 30 minut.

1. Teoretické základy informatiky

Výroková logika, jazyk, formule, pravdivost, vyplývání, tautologie. Booleovské funkce a funkčně úplné systémy. Úplné konjunktivní a disjunktivní normální formy. Základní syntaktické a sémantické pojmy výrokové logiky. Přehledově dokazatelnost ve výrokové logice. Syntax a sémantika predikátové logiky.

Množiny, množinové operace a vztahy, potenční množina, kartézský součin, číselné množiny. Relace, binární relace a jejich reprezentace, operace s relacemi. Funkce (zobrazení) a jejich vlastnosti. Binární relace na množině a jejich vlastnosti. Uzávěry binárních relací. Ekvivalence a rozklady. Uspořádání, Hasseovy diagramy, význačné prvky, svazy.

Konečné a nekonečné množiny, spočetné množiny, příklady, existence nespočetné množiny, diagonální metoda. Porovnání velikosti množin, Cantorova-Bernsteinova věta, Cantorova věta. Indukce a rekurse, matematická indukce a její varianty, strukturální indukce.

Pravidlo součtu a součinu, permutace, variace, kombinace. Binomická věta. Princip inkluze a exkluze.

Pravděpodobnost, Laplaceova definice pravděpodobnosti, pravděpodobnostní prostor, vlastnosti pravděpodobnosti. Náhodná veličina, střední hodnota.

Orientované a neorientované grafy, izomorfismus, podgrafy, pojmy k cestování. Hledání nejkratší cesty, Dijkstrův algoritmus. Minimální kostra grafu, Kruskalův algoritmus. Stromy, kořenové stromy, vztahy mezi výškou, počtem vrcholů a počtem listů.

Čísla a číselné obory. Vybrané číselné funkce a rychlosti jejich růstu. Dělitelnost, prvočísla, věty o jednoznačnosti. Největší společný dělitel a nejmenší společný násobek. Euklidův algoritmus. Kongruence modulo n a její vlastnosti.

Orientované a neorientované grafy, izomorfismus, podgrafy, pojmy k cestování, souvislost, stupně vrcholů. Hledání nejkratší cesty, Dijkstrův algoritmus. Stromy a jejich vlastnosti. Minimální kostra grafu, Kruskalův algoritmus. Kořenové stromy, vztahy mezi výškou, počtem vrcholů a počtem listů.

Matice, operace s maticemi, hodnost, determinant. Vektorové prostory, podprostory, báze a dimenze, matice přechodu. Eukleidovské vektorové prostory, ortogonální a ortonormální báze, Schwarzova nerovnost, Schmidtova ortogonalizační metoda. Soustavy lineárních rovnic, Frobeniova věta, Gaussova eliminační metoda, Cramerovo pravidlo. Lineární zobrazení a transformace a jejich matice.

Funkce jedné reálné proměnné, základní vlastnosti. Posloupnosti a jejich limity. Limita a spojitost funkce. Vlastnosti spojitých funkcí, spojitost složené a inverzní funkce.

Derivace funkce a její geometrický význam: Pravidla pro derivování funkcí, derivace složené funkce, derivace inverzní funkce, derivace elementárních funkcí. Průběh funkce: základní věty diferenciálního počtu, extrémy funkce, konvexní a konkávní křivky, asymptoty.

Neurčitý integrál a metody jeho výpočtu. Riemannův určitý integrál: definice, základní věta integrálního počtu, metody výpočtu. Geometrická interpretace určitého integrálu.

Formální jazyky a jejich hierarchie. Regulární jazyky (definice, uzávěrové vlastnosti). Konečné automaty deterministické a nedeterministické. Regulární výrazy, automaty s ϵ -přechody. Minimalizace konečného deterministického automatu. Pumping lemma. Bezkontextové jazyky a jejich vlastnosti (uzávěrové vlastnosti, jednoznačnost). Zásobníkové automaty deterministické a nedeterministické.

Turingův stroj (TS), nedeterministický TS. Jazyk přijímaný TS, jazyk rozhodovaný TS. Church-Turingova teze, varianty TS. Částečně rekurzivní a rekurzivní jazyky, jazyky a rozhodovací problémy. Vztah rekurzivních a částečně rekurzivních jazyků. Uzávěrové vlastnosti jazyků TS. Riceova věta.

Složitost algoritmu (časová a paměťová). Třída P, třída NP, důvody jejich zavedení, jejich vzájemný vztah. NP-úplné problémy. Cook-Levinova věta. Příklady NP-úplných problémů, dokazování NP-úplnosti. Třída PSPACE, její vztah k třídám P a NP, PSPACE-úplné problémy.

Algoritmus, problém, časová složitost algoritmu v nejhorším a průměrném případě. O-notace a růst funkcí, definice, vlastnosti, příklady použití. Problém třídění, rozdělení třídících algoritmů, dolní mez složitosti třídění porovnáváním. Základní metody třídění: insert sort, select sort, bubble sort. Quick sort a jeho složitost. Merge sort a jeho složitost. Heap sort a jeho složitost. Další metody třídění: counting sort, radix sort, bucket sort. Vnější třídění. Pořádkové statistiky.

Lineární datové struktury: seznam, zásobník, fronta. Vyhledávání v lineárních datových strukturách. Binární vyhledávací stromy, operace a jejich složitost. AVL stromy, operace a jejich složitost. B stromy, operace a jejich složitost. Hashovací tabulky, metody řešení kolizí. Základní grafové algoritmy: průchod do šířky, průchod do hloubky, topologické uspořádání.

2. Programování a vývoj software

Přehled a základní rysy programovacích paradigmat: funkcionální, procedurální, objektové. Symbolické výrazy a jejich vyhodnocování v jazyce Common Lisp. Rekursivní funkce a rekursivní výpočetní proces. Funkce vyššího řádu. Lexikální a dynamický rozsah platnosti proměnných. Makra. Líné vyhodnocování v datových strukturách, přísliby a proudy. Líné vyhodnocování v programovacích jazycích, aplikativní a normální model vyhodnocení. Zásobníkový model vyhodnocování.

Objektově orientované programování: třídy a objekty, zprávy a metody. Zapouzdření, polymorfismus, rozhraní, dědičnost (jednoduchá i vícenásobná). Pravidlo *is-a*, princip B. Liskovové, kontraktové programování. OOP založené na prototypch. Příklad jednoho nebo více objektově orientovaných jazyků (Java, C#, C++, Common Lisp, Python ...) a jejich objektově orientované rysy.

Paralelní program: procesy, stav, historie. Plánovač. Atomické operace. Programová logika, invarianty programu. Problém kritické sekce, Petersonův a Dekkerův algoritmus. Semafory: operace, invarianty, implementace. Výrobci a spotřebitelé, večerící východní mudrci, čtenáři a písaři. Bariéry, paralelní součet prefixů. Logické programování: pravidla, cíl, unifikace, úsudek, strom úsudků. Aritmetika v logickém programování.

Softwarový proces, jeho fáze a jeho modely (vodopádový, evoluční, iterativní). Specifikace požadavků (druhy požadavků, metody zjišťování požadavků, diagram případů užití). Návrh systému a využití jazyka UML při návrhu (diagram tříd, sekvenční diagram, stavový diagram, diagram aktivit). Sestavení kvalitního kódu (formátování kódu, efektivní využívání deklarací, konvence zápisu funkcí a jejich parametrů, zásady pro psaní příkazů). Testování programu (metody testování, úrovně testování, kategorie chyb). Návrhové vzory a vzory architektur SW. Verze software a systémy pro správu verzí.

Relační model databáze: atributy, n -tice, relace, relační proměnné, charakteristické vlastnosti relací. Relační algebra: množinové operace, restrikce, projekce, přirozené spojení, přejmenování atributů, relační dělení. **SELECT** výraz v SQL. Kontrola integrity: primární a alternativní klíče, cizí klíče. Funkční závislosti, Boyceho–Coddova normální forma, normalizace. Pohledy v SQL. Agregace a řazení v SQL.

Počítačové sítě, jejich služby a architektury. Ethernet: přepínač, použití média, linkový rámec. Protokol IP: paket, adresy a podsítě, směrování. Protokoly TCP a UDP: spojení a řízení toku dat. Systém DNS. Aplikační služby a tvorba síťových aplikací. Bezpečnost počítačových sítí.

Architektury a princip činnosti počítače. Číselné soustavy. Binární logika, logické operace a jejich vlastnosti, funkce a jejich úpravy, logické obvody. Reprezentace čísel a znaků v počítači. Detekční a samoopravné kódy.

Operační systém, architektura, poskytovaná rozhraní. Vykonávání programu a proces překladu. Správa procesoru: procesy a vlákna, plánování jejich běhu, komunikace a synchronizace. Problém uváznutí, jeho detekce a metody předcházení. Správa operační paměti: segmentace, stránkování, virtuální paměť. Přídělování a uvolňování paměti v uživatelských procesech, garbage collector.

3. Povinně volitelné předměty specializace Programování a vývoj software

Okruh je vymezen následujícími povinně volitelnými předměty specializace, student si z nich před zkouškou vybere předměty alespoň za 9 kreditů.

KMI/JCS1 Jazyk C# 1 (3 kr.)

Základní vlastnosti a konstrukty jazyka (typování, hodnotové a referenční datové typy, platforma .NET; cykly, podmínky, ...) Objektově orientované rysy jazyka (properties, fields, metody, dědičnost, modifikátory přístupu, rozhraní, delegáti) Generické datové typy, systém výjimek. Tvorba grafického uživatelského rozhraní.

KMI/JCS2 Jazyk C# 2 (3 kr.)

Práce se soubory a souborovým systémem, XML a JSON, serializace, deserializace. LINQ. Základy paralelního programování — vlákna. Práce s relačními databázemi, Objektově relační mapování a EntityFramework. Tvorba webových stránek s ASP.NET MVC, WebAPI a OData. Reflexe.

KMI/JJ1 Jazyk Java 1 (3 kr.)

OOP v jazyce Java. (Třídy, objekty, atributy, rozhraní, rozsahy platnosti, polymorfismus, dědičnost.) Specifické třídy jazyka (např. výčtové typy, záznamy, výjimky, obalové třídy). Generické třídy a metody. Kolekce a práce s nimi. Lambda výrazy, funkční rozhraní, vnořené a anonymní třídy, Stream API. Organizace kódu, testování, dokumentace, překlad a provádění programu.

KMI/JJ2 Jazyk Java 2 (3 kr.)

Zpracování dat ve formátu XML, srovnání různých přístupů. Tvorba grafického uživatelského rozhraní, principy použité v grafických knihovnách. Práce se sítí, podpora standardních protokolů, komunikace protokolem TCP/IP. Principy práce s relační databází, typy dotazů, zpracování dat. Reflexe, anotace.

KMI/JP Jazyk Python (3 kr.)

Základní datové typy, mutabilita a imutabilita. Sekvence: unpacking, cykly. Dekorátory a jejich použití. Systém dunder metod a objektově orientované programování. Struktura projektu: balíčky, moduly a jejich importování. Protokol iterování a comprehensions. Testování: principy a možnosti. Dokumentace a typování a PEP8.

KMI/JCP Jazyk C++ (3 kr.)

Vstupy a výstupy přes streamy cin, cout. Objekty a třídy. Konstruktory třídy a destruktory třídy. Dědičnost tříd. Přetížení operátorů (operator overloading). Šablony (templates). Třída pro řetězce (string). Třída pro regulární výrazy (regex). Knihovna standardních šablon STL. Kontejnery vektor, množina, mapa. Streamy pro soubory.

KMI/TMAP Tvorba mobilních aplikací (4 kr.)

Struktura Android mobilní aplikace. Aktivita a její životní cyklus. Ukládání dat v Android aplikacích. Práce na pozadí v Android aplikacích. Rozhraní Android mobilních aplikací a jeho tvorba. Možnosti vývoje multiplatformních mobilních aplikací. Jazyk Kotlin. Základy vývoje aplikací pro iOS.

KMI/WEBA Webové aplikace (3 kr.)

Webové aplikace a přehled technologií používaných při jejich tvorbě. Architektura webové aplikace a problematika škálovatelnosti. Zpracování HTTP požadavků: předávání dat mezi webových a aplikačním serverem, příklady realizace. REST API: popis a příklady realizace. JavaScript na webovém frontendu a jeho možnosti. Technologie AJAX a její použití. Knihovna React: charakteristika, použití. Možnosti tvorby nativní aplikací pomocí webových technologií. Node.js: charakteristika, použití.