

# Bakalářské zkoušky (příklady otázek)

podzim 2017

## 1 Automaty a gramatiky (3 body)

1. Definujte *zásobníkový automat* a popište všechny složky definice.
2. Definujte přijímání jazyka prázdným zásobníkem a přijímání jazyka koncovým stavem.
3. Zkonstruuje *deterministický zásobníkový automat* přijímající jazyk  $L = \{u\#v \mid u, v \in \{a, b\}^*, |u| \geq |v|\}$ .

## 2 Algoritmy a datové struktury: minimální kostra (3 body)

1. Definujte minimální kostru (souvislého neorientovaného) grafu.
2. Popište libovolný algoritmus pro hledání minimální kostry.
3. Charakterizujte hrany, které lze přidat k podgrafu minimální kostry tak, že vznikne opět podgraf minimální kostry.

## 3 Databáze (3 body)

1. Stručně definujte význam pojmu *izolace* (isolation) v ACID transakčním modelu.
2. Způsob, kterým je implementována izolace transakcí v konkrétním databázovém systému, často způsobuje omezení jejich souběžného zpracování. SQL-92 definuje čtyři úrovně zajištění izolace (READ UNCOMMITTED, READ COMMITTED, REPEATABLE READS, SERIALIZABLE). Vyberte si libovolné dvě a stručně (nejlépe jednou větou) vysvětlete, jaké izolační vlastnosti zajišťují nebo naopak dovolují porušit.
3. Mějme tabulku uživatelů (uziv) se sloupci (id, jmeno, plat, vek). Tabulka obsahuje záznamy:

```
(1, "Franta", 40000, 42)
(2, "Pepa", 70000, 54)
(3, "Tonda", 100000, 63)
```

Dále uvažme dvě probíhající transakce. Pro jednoduchost si představme, že jednotlivé příkazy těchto transakcí byly postupně provedeny v pořadí, jaké je určeno v následujícím schématu. Výsledek každé operace SELECT je uveden pod dotazem v komentáři.

```
/* Transaction 1 */
```

```
SELECT plat FROM uziv WHERE vek =
    (SELECT MAX(vek) FROM uziv)
/* plat == 100000 */
UPDATE uziv SET plat = 50000 WHERE id = 1
```

```
/* Transaction 2 */
```

```
SELECT plat FROM uziv WHERE id = 1
/* plat == 40000 */
SELECT id FROM uziv WHERE vek =
    (SELECT MAX(vek) FROM uziv)
/* id == 3 */
UPDATE uziv SET plat = 120000 WHERE id = 3
COMMIT
```

```
SELECT AVG(plat) FROM uziv
/* AVG(plat) == 80000 */
COMMIT
```

Na základě příkazů v transakcích a uvedených hodnot určete, jakou úroveň izolace dle SQL-92 (tedy READ UNCOMMITTED, READ COMMITTED, REPEATABLE READS, nebo SERIALIZABLE) splňuje databázový systém, který transakce zpracoval. Vaše rozhodnutí stručně zdůvodněte.

## 4 Programovací jazyky: principy objektového návrhu (3 body)

Mějme „tradiční“ souborový systém se soubory a hierarchickými adresáři. Soubory i adresáře mají svoje jméno, vlastníka, skupinu a přístupová práva ala UNIX (práva `rwX` pro vlastníka, skupinu a ostatní) a základní atributy (velikost a čas modifikace). Kromě souborů a adresářů souborový systém podporuje i symbolické odkazy (jak na soubory, tak i na adresáře).

1. Navrhněte sadu tříd, které budou objektovým způsobem reprezentovat adresářovou strukturu tohoto souborového systému. Instance navržených tříd mají umožňovat získávat informace o souborech/adresářích, tj. získat jejich jméno a jméno včetně celé cesty od kořene souborového systému, práva, atributy, u adresářů získat seznam položek tvořící obsah adresáře. (Do návrhu ale nezahrnujte metody pro další manipulaci se souborovým systémem jako vytváření, mazání, čtení či zápis souborů.)

Při tvorbě tříd respektujte pravidla objektového návrhu. Čeká se návrh objektového modelu, tedy v tomto bodě není třeba psát těla metod či privátní atributy.

2. Do návrhu přidejte metodu `walk()` (včetně těla), která, pokud je zavolána na adresáři, projde do hloubky celý adresář včetně podadresářů a aplikuje na všechny navštívené soubory/adresáře *operaci*, která je jako parametr předána metodě `walk()`. Operace by měla být ideálně předávána jako *lambda výraz*. Ukažte zavolání metody `walk()` s operací, která vytiskne jméno souboru/adresáře.

Ve vašem řešení dejte pozor na možnou existenci symbolických odkazů na adresáře – operace se na každý navštívený element smí provést právě jednou.

Pro řešení úlohy si vyberte jeden z jazyků Java, C++, C#. Při hodnocení odpovědi nebudou uvažovány drobné syntaktické chyby, ale obecně by použité konstrukce měly odpovídat zvolenému programovacímu jazyku.

## 5 Architektura počítačů a operačních systémů (3 body)

Předpokládejte následující program zapsaný v jazyce C# (klíčové slovo `static` označuje globální proměnnou nebo globální funkci, datový typ `int` je 32-bitový celočíselný se znaménkem):

```
1 using System;
2
3 class Program {
4     static int a;
5     static int b;
6     static int c;
7
8     static void ComputeResult() {
9         c = a / b;
10    }
11
12    static void Main(string[] args) {
13        a = int.Parse(Console.ReadLine());
14        b = int.Parse(Console.ReadLine());
15        ComputeResult();
16        Console.WriteLine(c);
17    }
18 }
```

Dále předpokládejte nějaký 32-bitový procesor s obecnou registrovou architekturou a s podporou pro hardwarové celočíselné dělení – dostačuje nějaká obecná představa procesoru, jehož instrukční sada by rámcově odpovídala typickému návrhu takových procesorů. Tedy můžete využít instrukce nějaké reálné procesorové architektury (např. x86), ale stejně tak je plně dostačující, pokud si nějakou přiměřeně realistickou instrukční sadu vymyslíte sami.

1. Zapište v assembleru uvažovaného procesoru posloupnost instrukcí, do které by se nejspíše přeložil jediný řádek C# kódu ve funkci `ComputeResult` (tj. řádek číslo 9 celého programu). Adresy proměnných si libovolně vhodně zvolte. Pro každou použitou instrukci napište stručný popis jejího chování.
2. Pokud výše uvedený program spustíme a na standardní vstup zadáme:

```
20
0
```

tak vypíše:

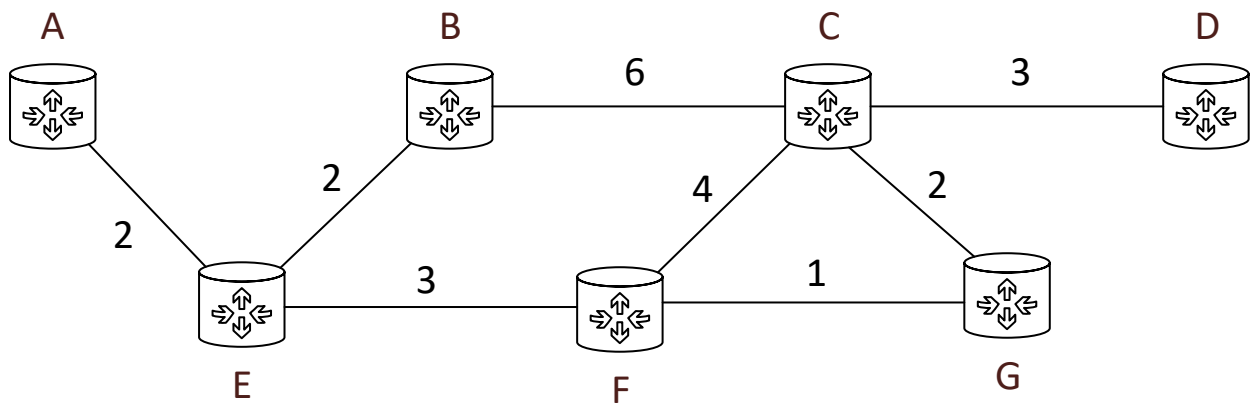
```
Unhandled Exception: System.DivideByZeroException: Attempted to divide by zero.  
  at Program.ComputeResult() in Program.cs:line 9  
  at Program.Main(String[] args) in Program.cs:line 15
```

Vysvětlete kdo (jaký kód) nejspíše vypisuje text uvedené chybové hlášky „Unhandled Exception ...“.

Vraťte se ke svému řešení první části otázky a vysvětlete na něm, jak je zařízeno, že když procesor bude vykonávat vámi uvedený strojový kód a dojde k pokusu o dělení nulou, tak dojde k výpisu zmíněné chybové hlášky. Stačí vyjmenovat a stručně popsat posloupnost jednotlivých hlavních celků operačního systému, uvedeného programu, atd., které se na ošetření dělení nulou budou v typické situaci podílet.

## 6 Základy sítí: směrování (3 body)

Mějme síť propojené směrovači (routery) dle obrázku. Routery jsou označeny písmeny, vzdálenosti k sousedním routerům jsou u hran grafu (hrany reprezentují síť spojující routery a vzdálenost představuje „pomalost“ daného spojení, tj. vyšší vzdálenost znamená pomalejší spojení). Routery na počátku znají pouze své sousedy a vzdálenosti k nim. K výpočtu směrovacích informací používají distribuovaný distance-vector algoritmus, který jako metriku používá součet vzdáleností mezi routery.



1. Jaké základní údaje musí obsahovat jednotlivé záznamy ve směrovací tabulce?
2. V kolikátém kroku algoritmu pro výpočet směrovacích informací znají pro síť na obrázku všechny routery úplnou optimální informaci o směrování? Počáteční stav je krok 0.
3. Napište obsah směrovací tabulky na routeru A po zjištění úplné informace.
4. Popište, jak budou reagovat ostatní routery, když vypadne router F. Změní to nějak vzdálenost z routeru A do routeru D?

## 7 Překladače (3 body)

Tato gramatika popisuje zjednodušenou verzi jazyka regulárních výrazů:

$$\begin{aligned} R &\rightarrow R \mid M \\ R &\rightarrow M \\ M &\rightarrow M B \\ M &\rightarrow B \\ B &\rightarrow E^* \\ B &\rightarrow E \\ E &\rightarrow \text{non-special-character} \end{aligned}$$

Počáteční neterminál je  $R$ ,  $\text{non-special-character}$  je libovolný znak různý od  $|$  a  $*$ .

1. Tato gramatika není LL(1) – vysvětlete proč (uveďte pokud možno všechny důvody).
2. Upravte gramatiku tak, aby generovala tentýž jazyk a byla přitom LL(1). Pojmenujte nebo stručně popište transformace, které jste při této úpravě použili.

## 8 Algoritmy a optimalizace: množinové pokrytí (otázka pro studijní zaměření – 5 bodů)

Množinové pokrytí je následující problém:

*Vstup:* Systém množin  $S_1, \dots, S_m \subseteq \{1, \dots, n\}$ , jejich ceny  $c_1, \dots, c_m \geq 0$ .

*Výstup:* Podsystem  $S_i, i \in I, I \subseteq \{1, \dots, m\}$  takový, že  $\bigcup_{i \in I} S_i = \{1, \dots, n\}$ .

*Cíl:* minimalizovat cenu  $I$ , tj.  $\sum_{i \in I} c_i$ .

Důležité parametry problému:  $g = \max_j |S_j|, f = \max_e |\{j \mid e \in S_j\}|$ .

1. Zformulujte lineární program, jehož celočíselné řešení a účelová funkce odpovídají přípustným řešením pro množinové pokrytí a jejich ceně.
2. Popište aproximační algoritmus založený na zaokrouhlování řešení lineárního programu z předchozího kroku. Odvoďte aproximační poměr tohoto algoritmu.
3. Popište hladový algoritmus pro množinové pokrytí a uveďte jeho aproximační poměr. Důkaz se nevyžaduje.

## 9 Algoritmy a optimalizace: Arrangement přímek (otázka pro studijní zaměření – 4 body)

1. Definujte arrangement přímek v rovině.
2. Popište (jakýkoliv) způsob reprezentace arrangementu přímek v rovině v počítači.
3. Navrhněte (jakýkoliv) polynomiální algoritmus, který zkonstruuje arrangement přímek v rovině ze zadané množiny přímek, a odhadněte co nejlépe jeho (asymptotickou) časovou složitost.

(Při popisu algoritmu je důležité popsat, z jakých dílčích kroků (popř. subalgoritmů) algoritmus sestává, ale není potřeba zacházet do podrobností ohledně implementace takových dílčích kroků.)

## 10 Diskrétní modely a struktury: barevnost (otázka pro studijní zaměření – 3 body)

1. Zformulujte Brooksovu větu o barevnosti grafů maximálního stupně  $\Delta$ .
2. Dokažte (bez použití Brooksovy věty), že je-li graf souvislý, má maximální stupeň  $\Delta$  a nelze ho obarvit  $\Delta$  barvami, pak je  $\Delta$ -regulární.
3. Nechť  $G$  je 3-regulární souvislý graf, který obsahuje dva vrcholy ve vzdálenosti větší než 2 od sebe. Ukažte použitím Brooksovy věty, že vrcholy  $G$  lze obarvit 9 barvami tak, aby žádné dva vrcholy ve vzdálenosti nejvýše 2 od sebe neměly stejnou barvu.

## 11 Diskrétní modely a struktury: kódy (otázka pro studijní zaměření – 3 body)

1. Najděte příklad samoopravného binárního kódu velikosti 4 se slovy délky 5 a minimální vzdáleností mezi slovy 3.
2. Ukažte, že existuje-li binární kód velikosti  $2n + 1$  se slovy délky  $k$  a min. vzdáleností  $d$ , pak také existuje binární kód velikosti  $n + 1$  se slovy délky  $k - 1$  a vzdáleností  $d$ .

## 12 Diskrétní modely a struktury: ordinály (otázka pro studijní zaměření – 3 body)

Definujte pojem ordinál (ordinální číslo).

Nalezněte tři nekonečné ordinály  $\alpha, \beta, \gamma$  takové, aby platily obě následující podmínky:

- $\alpha < \beta < \gamma$
- $|\alpha| = |\beta| < |\gamma|$

Zdůvodněte.

### 13 Matematická lingvistika: formální jazyky a automaty (otázka pro studijní zaměření – 3 body)

1. Vysvětlete pojem neprojektivity v závislostních stromech.
2. Uveďte příklad nějaké neprojektivní věty.
3. Vysvětlete, do jaké třídy jazyků podle Chomského hierarchie patří přirozené jazyky, které obsahují neprojektivní konstrukce.

### 14 Matematická lingvistika: základní formalismy pro popis přirozených jazyků (otázka pro studijní zaměření – 3 body)

1. Podrobně popište základní fakta o tvorbě jazykových korpusů. Uveďte základní charakteristiky, jak by měl moderní korpus vypadat.
2. Rozdělte jazykové korpusy podle typu značkování, pro každý typ uveďte alespoň dva příklady.
3. Uveďte příklady paralelních korpusů a vysvětlete jejich využití.

### 15 Matematická lingvistika: základy teorie informace (otázka pro studijní zaměření – 3 body)

1. Předpokládejme, že jednoduchý jazyk  $L$  obsahuje pouze 8 znaků ( $p, t, k, s, h, a, i, u$ ) a že tento jazyk je jejich náhodnou posloupností s následujícími pravděpodobnostmi:

$p$	$t$	$k$	$s$	$h$	$a$	$i$	$u$
$1/8$	$1/8$	$1/16$	$1/8$	$1/16$	$1/4$	$1/8$	$1/8$

Vypočítejte entropii znaků jazyka  $L$ .

2. Při podrobnějším zkoumání jazyka  $L$  bylo odhaleno používání slabik. Předpokládejme, že jazyk je náhodnou posloupností nezávisle generovaných slabik souhláska-samohláska s následujícími pravděpodobnostmi:

	$a$	$i$	$u$	
$p$	$1/8$	$1/8$	$0$	$1/4$
$t$	$1/8$	$1/16$	$1/16$	$1/4$
$k$	$1/16$	$0$	$1/16$	$1/8$
$s$	$1/8$	$1/16$	$1/16$	$1/4$
$h$	$1/16$	$0$	$1/16$	$1/8$
	$1/2$	$1/4$	$1/4$	

Dále pracujeme s nově zjištěnými pravděpodobnostmi podle uvedené tabulky. Tabulka pravděpodobností slabik může být interpretována jako sdružená pravděpodobnost dvou náhodných proměnných  $C$  a  $V$ :

- $C$ : slabika začíná souhláskou  $c \in \{p, t, k, s, h\}$
- $V$ : slabika končí samohláskou  $v \in \{a, i, u\}$

Vypočítejte vzájemnou informaci  $I(C; V)$ .

### 16 Počítačová grafika: matice (otázka pro studijní zaměření – 3 body)

Homogenní souřadnice a maticové transformace.

1. Jak se pomocí matic transformují objekty v 3D počítačové grafice?
2. Proč zavádíme homogenní souřadnice a matice  $4 \times 4$ ? Co nám umožňují realizovat?
3. Uveďte několik elementárních maticových transformací (translace, rotace, škálování – pokuste se matice napsat prvek po prvku) a jeden praktický příklad složené transformace (nemusíte konstrukci dotáhnout do konce, stačí naznačit postup).

## 17 Počítačová grafika: potlačování šumu (otázka pro studijní zaměření – 3 body)

Potlačování šumu v rastrovém obraze pořízeném digitálním fotoaparátem nebo kamerou.

1. Napište matematický model šumu; co předpokládáme, že senzor fotoaparátu nasnímal?
2. Uveďte jednoduché lineární metody redukce šumu.
3. Popište alespoň jeden nelineární algoritmus potlačující šum.

## 18 Počítačová grafika: stínování (otázka pro studijní zaměření – 3 body)

Lokální model odrazu světla na povrchu tělesa v simulované 3D scéně.

1. Popište jednoduchý lokální model odrazivosti (jak barva pozorovaná na povrchu tělesa závisí na poloze a intenzitě světelného zdroje).
2. Jak se takový model odrazivosti používá v realtime zobrazení (GPU, model složený z trojúhelníků, data jsou uložena ve vrcholech, GPU shadery)?
3. Jak je možné dosáhnout efektu hladkého tělesa, i když je 3D model sestaven z trojúhelníků (stačí popsat myšlenku, princip)?

## 19 Sítě a internetové technologie: IPv6 a NAT (otázka pro studijní zaměření – 3 body)

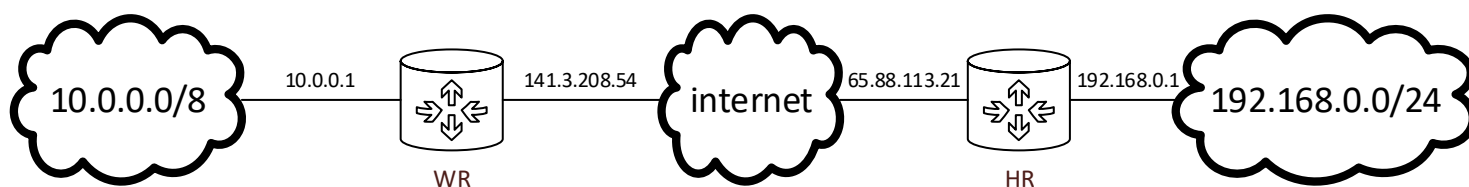
Vyjmenujte druhy IPv6 adres (není třeba znát prefixy) včetně zvláštních adres a vysvětlete jejich použití.

Jaké jsou důvody pro použití NAT v IPv4?

Používá se NAT i v IPv6 síti? Pokud ano, z jakého důvodu?

## 20 Sítě a internetové technologie: NAT (otázka pro studijní zaměření – 3 body)

Protože jste velmi výkonný pracovník (ale nejste administrátor sítě), tak máte v práci volný čas, který si chcete vyplnit sledováním filmů. Bohužel v práci nemáte žádné větší úložiště, kam byste si mohl(a) filmy uložit, zato doma je máte uložené na malém diskovém poli. Toto pole podporuje proprietární protokol pro přístup k uloženým souborům, kterému stačí, aby klienti měli přístup na TCP port 445 na diskovém poli. Váš počítač v práci je připojený do privátní sítě využívající NAT a má adresu 10.13.55.187. Ta samá situace je u Vás doma, tj. máte privátní síť za směrovačem připojeným k ISP, diskové pole má adresu 192.168.0.10. Situace odpovídá následujícímu obrázku:



1. Je nutné nějak nastavit domácí směrovač/firewall, aby bylo diskové pole dostupné z práce?
2. Je nutné nějak nastavit směrovač/firewall v práci, aby bylo diskové pole dostupné z práce?
3. Už máte oba směrovače správně nastavené (pokud to bylo vůbec zapotřebí). Popište všechny (bez průchodu neznámou částí sítě) změny adres a portů v TCP hlavičce, které nastanou při poslání paketu z klienta na domácí pole.

## 21 Sítě a internetové technologie: firewall a NFS (otázka pro studijní zaměření – 3 body)

Jako administrátor firmy jste konečně dokončil upgrade všech klientských strojů na novou verzi NFS klienta. Váš NFS server aktuálně používá verzi 3 a chcete přejít na verzi 4.1. Protože jste paranoidní správce, máte server důkladně zabezpečen firewallem a neběží tam žádné zbytečné služby.

1. Jak se změní nastavení firewallu při přechodu NFS serveru z verze 3 na verzi 4.1?
2. Je nutné nějak změnit (přidat, ubrat) služby běžící na NFS serveru při přechodu na verzi 4.1?
3. Protože jste opravdu paranoidní správce, rád byste zabezpečil(a) i samotnou komunikaci. Je to možné a pokud ano, jakým způsobem?

## 22 Systémové programování: dynamicky linkované knihovny (otázka pro studijní zaměření – 3 body)

Knihovnamí označujeme implementaci zpravidla tematicky souvisejících funkcí a datových struktur s rozhraním navrženým tak, aby dovolilo využívání knihovny v různých programech podle potřeby. Knihovny mohou být linkované staticky nebo dynamicky.

1. Vysvětlete hlavní rozdíl mezi statickým a dynamickým linkováním.
2. Uvažujte dvě verze stejného programu, které se liší pouze tím, že k jedné se knihovny linkují staticky a ke druhé dynamicky. Mohou se tyto verze programu lišit v rychlosti běhu nebo ve spotřebě paměti? Zdůvodněte proč.

## 23 Systémové programování: garbage collection (otázka pro studijní zaměření – 3 body)

1. Garbage collector používá koncept dosažitelnosti (reachability). Vysvětlete jak (formulujte dosažitelnost jako relaci a vysvětlete její použití).
2. V následujícím programu vyznačte, které objekty a ve kterém okamžiku nejdříve může garbage collector uvolnit z paměti. Předpokládejte, že program se spouští voláním metody `Main.main ()` s prázdnými argumenty.

```
1 class Data {
2     public Data neighbor;
3 }
4
5 public class Main {
6     public static void one (Data d) {
7         Data x = new Data ();
8         Data y = new Data ();
9         d.neighbor = x;
10        x.neighbor = y;
11        y.neighbor = x;
12    }
13    public static void two (Data d) {
14        Data a = new Data ();
15        Data b = new Data ();
16        Data c = new Data ();
17        a.neighbor = d;
18        b.neighbor = a;
19        c.neighbor = a;
20    }
21    public static void main (String [] arguments) {
22        Data d = new Data ();
23        one (d);
24        d.neighbor = null;
25        two (d);
26    }
27 }
```

Poznámka: Pro zápis kódu používá otázka základní konstrukce jazyka Java, nicméně tohoto jazyka se přímo netýká a použité konstrukce je možné vnímat obecně jako běžné konstrukce imperativního objektového jazyka. Případný kód v zápisu odpovědi může být zapsán podobně volně.

## 24 Systémové programování: návrhové vzory (otázka pro studijní zaměření – 3 body)

Vysvětlete použití návrhového vzoru singleton a načrtněte jeho implementaci (včetně podpory vláken).

Pro řešení úlohy si vyberte jeden z jazyků Java, C++, C#. Při hodnocení odpovědi nebudou uvažovány drobné syntaktické chyby, ale obecně by použité konstrukce měly odpovídat zvolenému programovacímu jazyku.

## 25 Databáze a web: programování na straně serveru (otázka pro studijní zaměření – 3 body)

Mějme webovou aplikaci, která je napsaná čistě v PHP bez použití skriptování na straně klienta. Veškerá interakce s uživatelem probíhá tedy pouze pomocí prvků dostupných v HTML (odkazy, formuláře). Aplikace si drží údaje o aktuální uživatelské relaci pomocí PHP sessions API, přičemž ID relace (session ID) je uloženo v cookies.

V aplikaci je (mimo jiné) zobrazována tabulka s položkami, u každé položky je tlačítko „Smazat“. Každé takové tlačítko je v samostatném formuláři, který se odesílá metodou POST (součástí odesílaných dat je jednoznačné ID položky). Stránku s tabulkou vyrenderuje skript `table.php` a POST požadavky generované jednotlivými mazacími tlačítky jsou obslouženy skriptem `delete.php`, který po smazání položky z databáze vrátí HTTP odpověď 303 (See Other) s hlavičkou `Location` obsahující URL vedoucí zpět na skript `table.php`.

Zákazník si přeje upravit aplikaci následovně: Po smazání položky je potřeba na stránce s tabulkou zobrazit zprávu „Záznam # byl úspěšně smazán.“, kde # představuje ID záznamu, a tlačítko „Beru na vědomí“, kterým uživatel potvrdí přečtení zprávy. Dokud není přečtení zprávy potvrzeno, musí se zpráva zobrazovat, i když uživatel obnoví (znovu načte) stránku, nebo se na stránku vrátí (např. pohybem o jednu stránku zpět a následně jednu stránku vpřed v historii prohlížeče). Naopak po potvrzení přečtení uživatelem se tato zpráva nesmí nikde zobrazit, ani když je obnovena stránka nebo uživatel použije funkce zpět/vpřed v prohlížeči. Celý mechanismus musí být připraven i na to, že uživatel má otevřeno více záložek s naší webovou aplikací a sdílí v nich uživatelskou relaci (cookie s uloženým session ID je platná pro všechny záložky s naší aplikací). Přesněji je potřeba zajistit, že se hláška zobrazí pouze v záložce, kde ke smazání došlo.

Popište (na podobné úrovni detailu, jako je zadání), jak bude požadovaná úprava implementována. Konkrétně co bude potřeba přidat do skriptů `table.php` a `delete.php` (případně do dalších vytvořených skriptů) a jaká (další) data budou kde uložena či předávána (v URL, v těle POST dotazu, v session, v cookies, ...). Při úpravě je potřeba zachovat koncept aplikace, která musí fungovat i v prohlížečích bez podpory Javascriptu.

Při řešení předpokládejte, že prohlížeč nedrží žádné dokumenty v cache a při obnově stránky nebo při pohybu zpět/vpřed v historii prohlížeče se vždy načte aktuální stránka.

## 26 Databáze a web: XPath (otázka pro studijní zaměření – 3 body)

Mějme XML soubor, ve kterém ukládáme data o zaměstnancích a pracovních skupinách. Zaměstnanec má nejvýše jednoho nadřízeného, může být členem libovolného počtu skupin a může být vedoucím libovolného počtu skupin. Skupiny navíc mohou obsahovat podskupiny (do libovolné hloubky). Náš dokument má následující DTD schéma:

```
<!ELEMENT zamestnanci (zamestnanec|skupina)*>
<!ELEMENT zamestnanec (jmeno, prijmeni, plat, vedouci*)>
<!ELEMENT jmeno (#PCDATA)>
<!ELEMENT prijmeni (#PCDATA)>
<!ELEMENT plat (#PCDATA)>
<!ELEMENT vedouci EMPTY>
<!ELEMENT skupina (clen+,skupina*)>
<!ELEMENT clen EMPTY>
<!ATTLIST zamestnanec id ID #REQUIRED nadrizeny IDREF #IMPLIED>
<!ATTLIST vedouci skupina IDREF #REQUIRED>
<!ATTLIST skupina id ID #REQUIRED nazev CDATA #IMPLIED>
<!ATTLIST clen id IDREF #REQUIRED>
```

Jednotlivé položky chápejte přirozeně (jejich název sémanticky odpovídá obsahu – tedy např. `plat` je celé číslo).

1. Napište XPath dotaz, který vrátí všechny skupiny, které mají alespoň 3 členy (vedoucí není člen skupiny).



- Napište XPath dotaz, který vrátí zaměstnance s nejvyšším platem.
- Napište XPath dotaz, který vrátí zaměstnance, kteří vedou strukturované skupiny (strukturované skupiny jsou takové, které mají podskupiny; podskupiny, které mají pod-podskupiny jsou také strukturované skupiny).

## 27 Databáze a web: index-sekvenční soubor (otázka pro studijní zaměření – 3 body)

1) Stručně popište organizaci index-sekvenčního souboru (ISF), který se používá k uložení DB tabulek s index-sekvenční přístupovou metodou (ISAM). Dále popište (v bodech) algoritmus nalezení záznamu v ISF dle jeho primárního klíče.

2) Mějme tabulku zaměstnanců, ve které je primární klíč příjmení. Tato tabulka je uložena v index-sekvenčním souboru, kde stupeň zaplnění bloků (blocking factor) v primární části je 2 a v indexové části 4.

V tabulce se nachází následující záznamy (pro jednoduchost uvádíme jen primární klíče): Bednárek, Tůma, Hric, Kruliš, Ježek, Zavoral, Yaghob, Bulej, Mráz, Pangrác, Mareš

Jak bude vypadat příslušný index-sekvenční soubor? Předpokládejte, že byl právě vytvořen ze všech záznamů najednou a že má být co nejvíce kompaktní. Výsledek zakreslete především z pohledu záznamů a bloků.

## 28 Logika (3 body)

- Uveďte definice (pro predikátovou logiku), kdy formule  $\varphi$  platí v teorii  $T$  (značeno  $T \models \varphi$ ), kdy je teorie  $T$  otevřená a kdy je teorie  $T$  splnitelná.
- Nechť  $L = \langle P, R \rangle$  je jazyk bez rovnosti, kde  $P, R$  je unární resp. binární relační symbol. Pomocí skolemizace následujících formulí či jejich negací nalezněte otevřenou teorii  $T$  nad vhodným rozšířením jazyka  $L$ , která je nesplnitelná, právě když

$$\{(\forall x)(\exists y)\neg R(x, y), (\forall x)(\exists y)(\forall z)(P(x) \vee R(y, z))\} \models (\forall x)P(x) \quad (1)$$

- Pomocí Vámi zvoleného dokazovacího systému (rezoluce, tablo metoda, Hilbertovský kalkül) dokažte, že  $T$  je nesplnitelná, anebo dokažte (1) přímo. (Nápověda: stačí dva rezoluční kroky.)

## 29 LA1 – Lineární závislost (3 body)

Definujte pojem lineární závislosti vektorů  $v_1, v_2, \dots, v_n$  alespoň dvěma způsoby.

## 30 LA2 – Projekce (3 body)

Ve vektorovém prostoru  $\mathbb{R}^3$  uvažme tři nenulové vektory  $x_1, x_2, x_3$ . Pro libovolný vektor  $v \in \mathbb{R}^3$  vezměme jeho projekci  $v_i$  na přímkou definovanou směrovým vektorem  $x_i$ ,  $i = 1, 2, 3$ . Rozhodněte, za jakých podmínek platí  $v = v_1 + v_2 + v_3$ . Odpověď zdůvodněte.

## 31 LA3 – Determinant nad $\mathbb{Z}_2$ (3 body)

Rozhodněte, zda nad dvouprvkovým tělesem (těleso  $\mathbb{Z}_2$  s nosnou množinou  $\{0, 1\}$ ) platí pro obecnou čtvercovou matici  $A$  a číslo (prvek tělesa)  $\alpha$ :

- $\det(\alpha A) = \alpha \det(A)$ ,
- $\det(-A) = \det(A)$ ,
- $\det(A^2) = \det(A)$ .

Odpovědi stručně zdůvodněte.

## 32 MA1 – Definice derivace (3 body)

Definujte pojem derivace funkce jedné reálné proměnné.

### 33 MA2 – Limita funkce (3 body)

Nalezněte hodnotu následující limity a výpočet zdůvodněte:

$$\lim_{x \rightarrow 0} (1 + \sin x)^{1/x}.$$

### 34 MA3 – Extrémy funkcí (3 body)

Buď  $f$  funkce dvou reálných proměnných daná předpisem  $f(x, y) = x^2 - 2y^3 + xy$ . Nalezněte lokální extrémy funkce  $f$  (a zdůvodněte, že se opravdu jedná o extrémy).

### 35 Rovinné grafy (3 body)

Rozhodněte, zda existuje souvislý rovinný graf s 2017 vrcholy, jehož všechny stěny (včetně vnější) jsou ohraničeny kružnicí délky 5. Odpověď samozřejmě zdůvodněte.

### 36 Kombinatorika – permutace (3 body)

Uvažme množinu permutací čísel  $\{1, 2, \dots, 10\}$ . Spočítejte, kolik z nich je takových, že:

1. nezobrazují žádné sudé číslo na sudé číslo,
2. nezobrazují žádné sudé číslo samo na sebe.

### 37 Grupy a podgrupy (3 body)

Nechť  $G$  je grupa s neutrálním prvkem  $e$ , binární operací  $\circ$  a operací inverzního prvku  $^{-1}$ . Definujte pojmy podgrupa a normální podgrupa grupy  $G$ . Jaký je vztah mezi těmito dvěma pojmy? (Vysvětlete, uveďte příklady.)