

## Maturitní otázky

### 1. Web a jeho realizace

- Internet × Webová stránka × HTML dokument
  - Webový prohlížeč
  - Webový server
  - Protokol HTTP
  - **Prakticky:** Popsat URL po částech s vysvětlením významu, popis odpovědi webového serveru (co se stane po zadání URL, než se zobrazí stránka; http metody)
- 

### 2. Web: HTML

- Jazyk HTML a jeho účel
  - HTML dokument × HTML element
  - Obsah HTML elementu × HTML atribut
  - Hypertextový odkaz
  - Párové a nepárové tagy
  - **Prakticky:** popis html kódu, doplnění do html kódu
- 

### 3. Web: CSS, formátování textu

- Jazyk CSS a jeho účel
  - Připojení CSS stylu k HTML dokumentu
  - CSS pravidlo × CSS selektor × CSS deklarace × CSS vlastnost
  - Způsob uplatňování CSS pravidel na HTML elementy
  - Příklady CSS vlastností na formátování textu
  - **Prakticky:** popsat stránku dle kódu (pozn.: hlavně styly)
- 

### 4. Web: JavaScript, základní programové konstrukce

- Jazyk JavaScript a jeho účel
  - Deklarace a inicializace proměnných
  - Výraz, příkaz, větvení, cykly
  - Definice a volání funkce
  - **Prakticky:** popsat kus javascript kódu
- 

### 5. Generické programování

- Význam generického programování
  - Generický typ **T (písmenko libovolné)**
  - Výhody generického programování
  - Nevýhody a omezení generického programování
  - Implementace a jazykové konstrukce
  - **Prakticky:** nastavit třídu, aby byla generická, popis ukázky kódu
- 

### 6. Abstraktní datové typy

- Abstraktní datové typy a jejich rozdíl od strukturovaných datových typů (obecně)
- Souvislost s rozhraním

- Souvislost s generickým programováním
  - Princip LIFO a FIFO
  - **Prakticky:** vyjmenování ADT, poznání konkrétního dle kódu
- 

7. Algoritmy, programovací jazyky a paradigmaty

- Algoritmus – definice
  - Vlastnosti algoritmu
  - Programovací jazyky x Skriptovací jazyky
  - Kompilace
  - Způsoby zápisu algoritmů (vývojový diagram, pseudokód)
  - Strukturované programování
  - Objektově orientované programování
  - **Prakticky:** na kódu vyjmenovat paradigmaty (OOP, strukturální, procedurální, funkcionální, event-driven...)
- 

8. Java: Principy strukturovaného programování

- Strukturované programování a jeho účel (důvod vzniku, přínosy)
  - Tři základní koncepty (jazykové konstrukce) strukturovaného programování
  - Logické výrazy (a konkrétně v Javě tři logické operátory, jejich priorita a chování)
  - **Prakticky:** napsat jazykové konstrukce, vyhodnotit logický výraz, napsat podmínku
- 

9. Java: Datové struktury, primitivní a strukturované datové typy

- Datové typy a jejich účel
  - Primitivní a strukturované datové typy-rozdíl mezi nimi (obecně)
  - Příklady konkrétních primitivních i strukturovaných datových typů v Javě
  - Hodnotové a referenční datové typy v Javě
  - **Prakticky:** deklarace a inicializace proměnných, vypsání programu
- 

10. Java: Řazení

- Řazení datových struktur
  - Souvislost s porovnáváním
  - Operace při řazení nad datovými strukturami
  - Významné řadící algoritmy
  - **Prakticky:** poznání řadícího algoritmu dle kódu, vysvětlení algoritmu, prohození prvků v poli
- 

11. Java: Principy objektově orientovaného programování

- Objektově orientované programování a jeho účel (důvody vzniku, přínosy)
  - Vztah mezi objektem a třídou
  - Atributy třídy
  - Základní principy objektově orientovaného programování
  - **Prakticky:** popsat na kódu principy tří OOP
- 

12. Java: Objekty, atributy a metody

- Objekt × Třída
- Jazyková konstrukce (zápis) definice třídy objektů v Javě
  - i. Konstruktor – *new*-výraz

- Výraz *static*
  - Modifikátory přístupu k atributům a metodám
  - Přetěžování konstruktorů
  - **Prakticky:** vytvoření definice třídy, přidání nějaké jednoduché metody pro práci s atributy (zkusit getery a setery)
- 

#### 13. Java: Zapouzdření

- Zapouzdření v souvislosti s Modifikátory přístupu
  - Zapouzdření v souvislosti s Algoritmy, Datovými strukturami a abstraktními datovými typy
  - Bezpečnostní význam zapouzdření
  - Metody pro nastavení/získání zapouzdřených dat – implementace
  - Zapouzdření a modifikátory přístupu v souvislosti s dědičností
  - **Prakticky:** Připravená třída, pomocí zapouzdření doplnit metody pro práci s atributy, nastavení přístupu atributů
- 

#### 14. Java: Dědičnost

- Dědičnost v souvislosti se vztahem mezi objekty Předka a Potomka
  - Omezení dědičnosti (počet předků, souvislost s rozhraním)
  - Jazykové konstrukce (zápis) definice třídy Potomek odvozené od třídy Předek – extends, super
  - Přístupová práva dědičnosti
  - **Prakticky:** Kód třídy, doplnit potomka, dědičnost v Javě u významných třídy (Výjimky, GUI)
- 

#### 15. Java: Polymorfismus

- Polymorfismus v souvislosti se vztahem mezi objekty Předka a Potomka
  - Jazyková konstrukce (zápis) přepsání metody Předka ve třídě Potomek
  - Rozdíl mezi přepsáním (redefinování) a přetížením metody
  - Rozhraní a souvislost s přepsáním metody
  - Souvislost rozhraní s abstraktními datovými typy
  - **Prakticky:** implementace rozhraní, popsání významných rozhraní v Java (List, Comparable, ActionListener...)
- 

#### 16. Java: Výjimky

- Význam chyby syntaxe a význam logické chyby
  - Rozdíl Error x Exception
  - Významné výjimky a významné chyby
  - Rozdíl mezi throws x throw
  - Ošetřování výjimek
  - **Prakticky:** kód – zahození výjimky, odchycení výjimky, vztah mezi významnými výjimkami
- 

#### 17. Java: Vyhledávání a porovnávání

- Účel vyhledávání (V čem spočívá úloha „najít hodnotu v datové struktuře“?)
- Příklad algoritmu, který vyhledává hodnotu v posloupnosti hodnot

- Souvislost porovnávání a vyhledávání
  - Jak souvisí rozdíl (odčítání) a výsledek s porovnáváním
  - Souvislost s rozhraním – přirozené řazení (Comparable), absolutní řazení (Comparator)
  - **Prakticky:** třída `dána`, nastavení možnosti porovnávání, volání metody pro využití komparátoru
- 

18. Java: Práce se soubory

- Účel třídy `File`
  - Kódování znaků a možné problémy při čtení (parsování)
  - Význam bufferování
  - Appending – význam při psaní do souboru
  - Významné operace při čtení a psaní do souborů
  - **Prakticky:** Představení možnosti čtení textu ze souboru, velikost souboru, popis kódu
- 

19. Java: GUI, prvky, jejich vlastnosti a události

- Rozdíl mezi grafickým uživatelským rozhraním (GUI) a rozhraním příkazového řádku (CLI)
  - Příklady prvků GUI a jejich vlastností
  - Příklady událostí v rámci GUI
  - Princip událostmi řízeného programování
  - **Prakticky:** popsání GUI dle kódu, doplnění funkcionality
- 

20. Java: GUI, vizuální rozvržení prvků

- Rozložení prvků – rozhraní `Layout`
- Prvky typu `Container`
- Okno bez nastaveného `Layout`
- Okno s `Layout` nastaveným na `null`
- Práce s více okny – implementace
- **Prakticky:** nalezení bodu dle souřadnic, popsání `layoutů` a prvků na obrázku