



# Záverečný test

## Zadanie



Ústav informatiky  
Prírodovedecká fakulta  
UPJŠ v Košiciach

Dvakrát meraj (rozmýšľaj), raz rež (programuj)

**Dôležité pravidlá a informácie** (viac na stránke predmetu):

- čas na riešenie úloh je **240 minút**,
- nie je dovolená žiadna (elektronická aj neelektronická) komunikácia s kýmkoľvek okrem dozoru
- v prípade akýchkoľvek problémov alebo z dôvodu ohodnotenia riešenia kontaktujte dozor,
- riešenia je možné nechať si ohodnotiť aj priebežne (nie až v závere testu),
- **funkčnosť každej metódy musí byť preukázaná spustením na vami vytvorenom testovacom vstupe, nespustiteľné metódy neumožňujú zisk príslušných bodov,**
- všetky inštančné premenné musia byť neverejné.

## Obecné/Mestské zastupiteľstvo

**Motivácia:** Štruktúra obecných a mestských zastupiteľstiev je skoro rovnaká bez ohľadu na to, či sa pozrieme do Bratislavy, Košíc, Prešova, Spišskej Novej Vsi, Fekišoviec alebo Bardejova. Body rokovania v jednotlivých zastupiteľstvách sa už samozrejme líšia podľa potrieb a problémov, ktoré samosprávy riešia. Našou úlohou je vytvoriť systém na správu zasadaní mestského zastupiteľstva, ktorý uchováva informácie o všetkých vystúpeniach.



**Pohľad analytika:** Pri implementácii aplikácie budeme potrebovať:

- triedu **Vystupenie**, ktorá reprezentuje jedno vystúpenie v zastupiteľstve, a
- triedu **Rozprava**, ktorá uchováva zoznam vystúpení v rámci rozpravy.

**Zadanie:** V balíku `sk.upjs.finalterm` vytvorte triedu **Vystupenie** obsahujúcu dátové položky prístupné cez `gettre` (a podľa uváženia aj modifikovateľné cez `settre`):

- **bodRokovania** (názov bodu rokovania, ktorého sa vystúpenie týka, napr. „Parkovacie miesta pre študentov UPJŠ“),
- **rečník** (vystupujúci rečník, napr. „Janko Hraško“),
- **datum** (dátum vystúpenia vo formáte DD.MM.RRRR, napr. „19.12.2018“ alebo „2.1.2019“),
- **cas** (čas začiatku vystúpenia vo formáte HH:MM:SS, napr. „15:20:30“),
- **trvanie** (trvanie vystúpenia v sekundách),
- **typ** (typ vystúpenia je buď „diskusny prispevok“, „fakticka poznamka“ alebo „interpelacia“),
- **relevantnaOsoba** (meno osoby – predrečníka, ktorej sa dané vystúpenie týka - iba ak je typ „fakticka poznamka“ alebo „interpelacia“).

*Poznámka:* K jednému bodu rokovania môže mať jeden rečník najviac jeden diskusný príspevok a jednu interpeláciu. K jednému diskusnému príspevku (ktorý je určený bodom rokovania a rečníkom) môže mať reagujúci rečník najviac jednu faktickú poznámku. Tieto pravidlá zabezpečujú, že každé vystúpenie je jednoznačne určené bodom rokovania, rečníkom a relevantnou osobou. Diskusný príspevok, ktorého sa týka faktická poznámka, vieme určiť jednoznačne na základe bodu rokovania a rečníka diskusného príspevku, ktorý je rovnaký ako relevantná osoba faktickej poznámky.

*Upozornenie:* Zadanie pre triedu **Vystupenie** predpisuje dátové položky prístupné cez `gettre`. Aké privátne inštančné premenné použijete na uloženie týchto dátových položiek je na vašom rozhodnutí.

Ďalej vytvorte aj triedu `sk.upjs.finalterm.Rozprava`, ktorá bude uchovávať zoznam vystúpení.

### Konštruktory a pridávanie vystúpení do zoznamu (3 body dokopy – povinné):

- **public** Vystupenie(String bodRokovania, String recnik, String datum, String cas, **int** trvanie) – použije sa na vytvorenie záznamu o vystúpení, ktoré je diskusným príspevkom,
- **public** Vystupenie(String bodRokovania, String recnik, String datum, String cas, **int** trvanie, String typ, String relevantnaOsoba) – použije sa na vytvorenie záznamu o vystúpení, ktoré je faktickou poznámkou alebo interpeláciou.
- **public void** pridaj(Vystupenie vystupenie) – inštančná metóda v triede Rozprava, ktorá pridá záznam o vystúpení do rozpravy.

### Práca so súborami (povinné):

#### V triede Vystupenie:

- **public static** Vystupenie zoStringu(String popis) – statická metóda, ktorá vráti referenciu na novovytvorený objekt triedy Vystupenie. Parameter je String v tvare "bodRokovania \t recnik \t datum \t cas \t trvanie", resp. "bodRokovania \t recnik \t datum \t cas \t trvanie \t typ \t relevantnaOsoba ", ak vystúpenie nie je diskusný príspevok (3 body);

*Poznámka:* Znak \t je neviditeľný znak tabulátora. Scanner-u môžete povedať, že oddeľovač má byť tabulátor zavolaním jeho metódy useDelimiter("\t"). Medzera pred a za \t sú len kvôli zlepšeniu čitateľnosti zadania, v reťazci reálne nie sú.

- **public** String toString() – vráti reťazec vhodne reprezentujúci údaje o vystúpení (1 bod).

#### V triede Rozprava:

- **public static** Rozprava zoSuboru(String nazovSuboru) – statická metóda, ktorá z uvedeného súboru prečíta rozpravu (zoznam vystúpení), pričom v každom riadku bude popis práve jedného vystúpenia (4 body).
- **public void** uloz(String nazovSuboru) – uloží všetky vystúpenia rozpravy do súboru v tvare, ktorý vie spracovať metóda zoSuboru(String nazovSuboru) (3 body).
- **public** String toString() – vráti reťazec vhodne reprezentujúci všetky vystúpenia rozpravy (1 bod).

### Inštančné metódy triedy Rozprava:

Ak niektorá z metód nevie vrátiť referenciu na objekt s požadovanými vlastnosťami, metóda nech vráti **null**.

- **public int** vratDlzkvVystupeni() – vráti celkový čas trvania všetkých vystúpení v sekundách (1 bod).
- **public** Set<String> vratDatумыZastupitelstiev() – vráti množinu dátumov, kedy sa konali zastupiteľstvá (2 bodov).
- **public** Rozprava vratRozpravuZoDna(String datum) – vráti zoznam vstúpení (vo forme rozpravy), ktoré sa uskutočnili počas dňa určeného parametrom (2 body).
- **public int** priemernaDlзкаDiskusnehoPrispevku() – metóda vráti priemerné trvanie diskusných príspevkov v sekundách (3 body).
- **public** List<String> vratInterpelovaneOsoby(String bodRokovania) – vráti zoznam všetkých mien relevantných osôb v interpeláciách, pričom každé meno je v zozname uvedené len raz (4 body).
- **public** List<Vystupenie> vratVystupeniaPodlaTypu(String recnik, String typ) – vráti tie vystúpenia rečníka, ktoré zodpovedajú typu zadanému parametrom typ. Ak je typ prázdny reťazec alebo **null**, tak metóda vráti všetky rečníkove vystúpenia (4 body).
- **public** List<String> vratNeaktivnychZastupcov(List<String> zastupcovia) – parametrom zastupcovia je zadaný zoznam všetkých zástupcov (napr. obecných poslancov). Metóda vráti

zoznam mien zvolených zástupcov, ktorý nemali ani jedno vystúpenie, pričom každé meno je v zozname uvedené len raz (3 body + 4 body, ak sa zoznamom vystúpení bude iterovať len raz).

- **public** Map<String, Integer> dlzkaVystupeniKBodomRokovania() – vráti mapu bodov rokovania a času v sekundách, koľko trvali vystúpenia k danému bodu rokovania (6 bodov).
- **public** Set<String> vratOpakovaneBodyRokovania() – vráti množinu bodov rokovania, ku ktorým sú vystúpenia aspoň v dvoch rôznych dátumoch (5 bodov + 4 body za implementáciu, ktorá zoznamom vystúpení preiteruje len raz).
- **public** Rozpravu vratVystupeniaVObdobi(String odDatumu, String poDatumu) – vráti referenciu na novovytvorený zoznam vystúpení (vo forme rozpravy), ktoré sa uskutočnili medzi dátumami odDatumu a poDatum (vrátane). Vzájomné poradie záznamov musí byť zachované (6 bodov).
- **public** String vratNajstrucnejsiehoRecnika(int minPocetVystupeni) – vráti meno rečníka, ktorý mal aspoň zadaný počet vystúpení (minPocetVystupeni) a zároveň priemerná dĺžka jeho vystúpenia bola najmenšia – ak je takých viac, vráťte ľubovoľného z nich (8 bodov).
- **public** String vratNajdiskutovanejsiBodRokovania() – vráti bod rokovania, ku ktorému vystúpilo najviac rôznych rečníkov – ak je takých viac, vráťte ľubovoľný z nich (8 bodov).

### Triedenie a komparátor (dokopy 5 bodov):

Vytvorte triedu VystupenieKomparator implementujúcu rozhranie `java.util.Comparator<Vystupenie>` s metódou (3 body):

- **public int** compare(Vystupenie vystupenie1, Vystupenie vystupenie2) – porovná vystúpenia chronologicky podľa dátumu a času.

V triede Rozprava implementujte inštančnú metódu (2 body):

- **public void** usporiadaj() – usporiada vystúpenia v zozname pomocou predchádzajúcej metódy.

### Výnimky (2 body)

Vytvorte nekontrolovanú výnimku `NeznamyTypException` a vhodne ju použite aspoň v jednej metóde ako indikátor, že reťazec nie je jedným z 3 povolených typov vystúpenia.

### Vylepšenia:

- (5 bodov) Upravte metódu `pridaj` v triede `Rozprava` tak, aby skontrolovala, či:
  - Pridávané vystúpenie má dátum a čas začiatku neskorší ako vystúpenia v rozprave
  - Ak ide o interpeláciu alebo faktickú poznámku, tak rečník, na ktorého sa reaguje, už vystúpil v rámci daného bodu rokovania.
- (6 bodov) Upravte svoje riešenia tak, aby korektne fungovali aj v prípade, keď reťazce s dátumami nie sú normalizované. Teda po úprave by reťazce “01.01.2019”, “1.01.2019”, “01.1.2019”, či “1.1.2019” mali reprezentovať rovnaký dátum. Napríklad môžete zvážiť vytvorenie pomocnej metódy, ktorá normalizuje dátum do jednotného formátu.

**Rada:** Ak chcete ako oddeľovač (delimiter) tokenov pre `Scanner` nastaviť bodku, použite `useDelimiter("\\.")`