



# Záverečný test

## Zadanie



Dvakrát meraj (rozmyšľaj), 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.

## Výmenná burza

**Motivácia:** Výmenný obchod je jedným z najstarších druhov obchodu. Vynájdenie peňazí zredukovalo množstvo výmenných obchodov, ale aj dnes sa môžeme stretnúť s výmenným obchodom - od detí predškolského veku až po obrovské medzištátne výmeny tovarov a služieb. Veľmi bežné sú výmenné burzy poštových známok, hokejových kartičiek alebo povianočné výmeny darčiekov. Našou úlohou je vytvoriť systém, ktorý pomôže uchovávať informácie a výmenách a poskytne funkcionalitu na analýzu vykonaných výmen.



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

- triedu `Vymena`, ktorá reprezentuje jednu uskutočnenú výmenu (výmenný obchod),
- triedu `ZoznamVymen`, ktorá bude uchovávať zoznam uskutočnených výmen.

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

- `ucastnik1` (meno prvého účastníka výmenného obchodu),
- `ucastnik2` (meno druhého účastníka výmenného obchodu),
- `predmet1` (názov predmetu, ktorý mal pred výmenou prvý účastník a získal ho druhý účastník)
- `predmet2` (názov predmetu, ktorý mal pred výmenou druhý účastník a získal ho prvý účastník)
- `stav1` (popis stavu predmetu `predmet1`)
- `stav2` (popis stavu predmetu `predmet2`)
- `cas` (čas v sekundách od začiatku konania burzy kedy prebehla výmena),
- `cena1` (odhadovaná cena predmetu `predmet1` v eurách),
- `cena2` (odhadovaná cena predmetu `predmet2` v eurách).

*Poznámka:* `stav1` a `stav2` nadobúdajú iba jednu z nasledujúcich hodnôt „neotvorený“, „nový“, „pouzivany“ alebo „poskodený“.

*Upozornenie:* Zadanie pre triedu `Vymena` 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.ZoznamVymen`, ktorá bude uchovávať zoznam výmen.

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

- **public** Vymena(String ucastnik1, String ucastnik2, String predmet1, String predmet2, String stav1, String stav2, **int** cas) – použije sa na vytvorenie záznamu o výmene, v ktorej účastníci nemali záujem o zverejnenie alebo odhadnutie cien,
- **public** Vymena(String ucastnik1, String ucastnik2, String predmet1, String predmet2, String stav1, String stav2, **int** cas, **double** cena1, **double** cena2) – použije sa na vytvorenie záznamu o výmene, v ktorej účastníci odhadli a zverejnili cenu vymieňaných predmetov,
- **public void** pridaj(Vymena vymena) – inštančná metóda v triede ZoznamVymen, ktorá pridá záznam o výmene do zoznamu výmen.

### Práca so súbormi (povinné):

V triede Vymena:

- **public static** Vymena zoStringu(String popis) – statická metóda, ktorá vráti referenciu na novovytvorený objekt triedy Vymena. Parameter je **String** v tvare "ucastnik1 \t ucastnik2 \t predmet1 \t predmet2 \t stav1 \t stav2 \t cas", resp. "ucastnik1 \t ucastnik2 \t predmet1 \t predmet2 \t stav1 \t stav2 \t cas \t cena1 \t cena2", ak výmena má odhady cien (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 výmene (1 bod).

V triede ZoznamVymen:

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

### Inštančné metódy triedy ZoznamVymen:

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

- **public int** vratPocetVymenSCenou() – vráti koľko výmen má odhad ceny (1 bod).
- **public** ZoznamVymen vratVymenyUcastnika(String ucastnik) – vráti zoznam výmen, v ktorých bol jedným z účastníkov ucastnik (2 body).
- **public** List<String> vratUcastnikov() – vráti zoznam všetkých mien účastníkov výmen, pričom každé meno je v zozname uvedené len raz (3 body).
- **public double** vratOdhadCeny(String predmet, String stav) – metóda vráti minimálnu cenu zo všetkých predmetov, ktoré majú rovnaký názov a stav ako sú zadané parametrami metódy (3 body).
- **public int** vratPocetVymenSCenovymNesuladom(**double** percento) – vráti počet výmen s odhadovanými cenami, kde je percentuálny cenový rozdiel medzi drahším a lacnejším predmetom aspoň percento percent (4 body).
- **public** List<Vymena> vratVymenySoStavmi(String stav1, String stav2) – vráti tie výmeny, v ktorých má jeden z predmetov stav1 a ďalší predmet stav2. Ak je stav2 prázdny reťazec alebo **null**, tak metóda vráti výmeny, kde je aspoň jeden zo stavov stav1 (4 body).
- **public int[]** vratHistogramPoHodinach() – vráti pole, ktoré na indexe h bude obsahovať počet výmen, ktoré v danú hodinu h od začiatku burzy prebehli. (4 body).

- **public** String najoblubenejsiPartner(String ucastnik) – vráti meno účastníka, ktorý sa zúčastnil najväčšieho počtu výmen s účastníkom ucastnik (5 bodov).
- **public** Set<String> vratStarinarov() – vráti množinu mien účastníkov, ktorí vždy vymieňali iba používaný a poškodený predmet (5 bodov + 4 body za implementáciu, v ktorej sa zoznam výmen bude iterovať len raz).
- **public** String najziskovejsiUcastnik() – vráti meno účastníka, ktorý pri výmenách ohodnotených predmetov dosiahol najväčší zisk, za zisk alebo stratu pri výmene predmetov považujeme rozdiel medzi odhadovanými cenami (5 bodov).
- **public** Map<String, Integer> casStravenyNaBurze() – vráti mapu účastníkov a času v sekundách, ktorý strávili na výmennej burze. Predpokladáme, že účastník bol na burze aspoň 5 minút pred svojou prvou výmenou a aspoň 5 minút po svojej poslednej výmene (7 bodov).
- **public** List<String> cestaCezPriekupnikov(String ucastnik, String predmet, String stav) – v tejto úlohe predpokladáme, že každý predmet sa dá jedinečne identifikovať na základe reťazcov predmet a stav. Metóda vráti postupnosť majiteľov predmetu (predmet je identifikovaný na základe parametrov predmet a stav) od jeho prvého majiteľa (parameter ucastnik) až po posledného podľa toho, ako sa daný predmet postupne vymieňal v čase. Hint: ak sú výmeny usporiadané podľa času tak nemusíte prechádzať zoznamom výmen opakovane (8 bodov).
- **public** String vratNajznamejsiehoUcastnika() – za najznámejšieho účastníka považujeme toho, ktorý si vymieňal predmety s najväčším počtom rôznych účastníkov (8 bodov).

### Triedenie a komparátor (dokopy 3 body):

Vytvorte triedu CasVymenyKomparator implementujúcu rozhranie `java.util.Comparator<Vymena>` s metódou (1 bod):

- **public int** compare(Vymena vymena1, Vymena vymena2) – porovná výmeny chronologicky podľa času od začiatku burzy.

V triede ZoznamVymen implementujte inštančnú metódu (2 body):

- **public void** usporiadajChronologicky() – usporiada výmeny v zozname chronologicky.

### Výnimky (2 body)

Vytvorte nekontrolovanú výnimku `NeznamyStavException` a vhodne ju použite aspoň v jednej metóde.

### Tipy:

- pridajte to triedy `Vymena` pomocnú metódu, ktorá vám vráti, či výmena obsahuje odhady cien
- zväzte vytvorenie metódy v triede `Vymena`, ktorá vytvorí výmenu, v ktorej sú účastníci vymenení