



# Záverečný test

## Zadanie



Ústav informatiky  
Prírodovedecká fakulta  
UPJŠ v Košiciach

Dvakrát meraj (rozmýšľaj), raz rež (programuj)

### Pravidlá a informácie:

- 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,
- funkčnosť každej metódy musí byť preukázaná spustením na vami vytvorenom testovacom vstupe, nespustiteľné metódy neumožňujú zisk príslušných bodov,**
- všetky inštančné premenné a metódy s výnimkou metód uvedených v zadaní musia byť neverejné.

## Internetová reklama

**Motivácia:** Bez reklamy by obchod len ťažko fungoval. To isté platí aj vo svete internetu, ktorého neoddeliteľnou časťou je internetová reklama – odkazy, ktoré sa zjavujú na rôznych stránkach a po kliknutí na ktoré sme presmerovaní na stránku reklamovaného produktu. Tieto reklamné prvky sa nazývajú (aj) reklamné boxy. Existuje veľa spôsobov predaja internetovej reklamy. Jeden z nich je ten, že klient si objednáva určitý počet zobrazení svojho reklamného boxu (alebo obrázkového baneru) na stránkach, s ktorými má reklamná agentúra dohodu a prípadne si pripláca za to, že na reklamu niekto klikol. Vašou úlohou je naprogramovať si vlastný systém na spravovanie reklám.



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

- triedu *ReklamnyBox*, ktorá bude uchovávať údaje (text, štatistika zobrazení, ...) o jednom reklamnom boxe (odkaze) s určitým predplateným počtom zobrazení. Box je **zobrazovaný**, ak predplatený počet zobrazení je väčší ako doposiaľ realizovaný počet zobrazení.
- triedu *Objednavky*, ktorá bude uchovávať zoznam všetkých objednaných reklamných boxov a poskytovať údaje webovej aplikácii realizujúcej samotné zobrazovanie reklám na stránkach.

**Zadanie:** V balíku *sk.upjs.finalTerm* vytvorte triedu *ReklamnyBox* obsahujúcu nasledujúce premenné:

- zadavatel** (firma zadávajúca reklamu, napr. "UPJŠ"),
- predplateneZobrazenia** (predplatený počet zobrazení reklamy, napr. 50000),
- text** (reklamný text, napr. "PAZ1a – naprogramujte si lepšiu kariéru"),
- url** (adresa stránky, na ktorú smeruje reklamný text, napr. "http://web.ics.upjs.sk/paz/"),
- pocetZobrazeni** (počet doposiaľ realizovaných zobrazení reklamy, napr. 30000),
- pocetKliknuti** (počet kliknutí na reklamu návštevníkmi stránky, napr. 130).

Ďalej vytvorte triedu *sk.upjs.finalTerm.Objednavky*, ktorá bude uchovávať zoznam všetkých objednaných (zobrazovaných aj nezobrazovaných) reklamných boxov.

### Konštruktory a pridávanie reklamných boxov (4 body dokopy – povinné):

- public** *ReklamnyBox*(String *zadavatel*, **int** *predplateneZobrazenia*, String *text*, String *url*, **int** *pocetZobrazeni*, **int** *pocetKliknuti*) – použije sa na vytvorenie záznamu o reklamnom boxe, ktorý už bol zobrazovaný
- public** *ReklamnyBox*(String *zadavatel*, **int** *predplateneZobrazenia*, String *text*, String *url*) – použije sa, ak ide o nový reklamný box, ktorý nebol ešte ani raz zobrazovaný
- public void** *pridaj*(*ReklamnyBox* *box*) – inštančná metóda v triede *Objednavky*, ktorá pridá reklamný box do zoznamu objednaných reklamných boxov

## Práca so súbormi (odporúčané):

### V triede *ReklamnyBox*:

- **public static** *ReklamnyBox* zoStringu(String popis) - statická metóda, ktorá vráti referenciu na novovytvorený objekt triedy *ReklamnyBox*. Parameter je *String* v tvare "zadavatel \t predplatene zobrazenia \t text \t url \t pocet zobrazeni \t pocet kliknuti" ak reklamný box už bol zobrazovaný, resp. "zadavatel \t predplatene zobrazenia \t text \t url" ak nebol. (4 body)  
Pozn: Znak \t je neviditeľný znak tabulátora. *Scanner*-u môžete povedať že oddeľovač má byť tabulátor zavolaním jeho metódy *useDelimiter("\t")*.
- **public** *String* toString() - vráti reťazec vhodne reprezentujúci daný reklamný box (1 body)

### V triede *Objednavky*:

- **public static** *Objednavky* zoSuboru(File f) - statická metóda, ktorá z uvedeného súboru prečíta zoznam reklamných boxov, pričom v každom riadku bude popis jedného boxu (4 body).
- **public void** uloz(File subor) - uloží všetky záznamy o všetkých reklamných boxoch do súboru (3 body).
- **public** *String* toString() - vráti reťazec vhodne reprezentujúci všetky objednané reklamné boxy (1 bod).

## Inštančné metódy triedy *Objednavky* (definícia zobrazovaného boxu je v pohľade analytika):

- **public int** pocetZobrazovanychBoxov() - vráti počet zobrazovaných boxov (2 body).
- **public** *List*<*ReklamnyBox*> zoznamBoxovZadavateľa(String zadavatel) - vráti referenciu na novovytvorený zoznam reklamných boxov daného zadávateľa (3 body)
- **public** *ReklamnyBox* zobrazBox() - vráti ďalší box na zobrazenie, t.j. zobrazovaný box, ktorý má najmenší pomer uskutočnených zobrazení boxu k predplatenému počtu zobrazení, a zvýši mu počet zobrazení o 1. Ak žiaden taký box sa už nezobrazuje, vráti **null**. (5 bodov)
- **public void** simuluj(int pocetZobrazeni, double pravdepodobnostKliknutia) - s využitím metódy *zobrazBox* odsimuluje zobrazenie *pocetZobrazeni* reklamných boxov pričom číslo *pravdepodobnostKliknutia* (0-100) vyjadruje v percentách pravdepodobnosť, s ktorou čitateľ stránky klikol na zobrazený reklamný box (5 bodov)
- **public int** zostavaZobrazeni(String zadavatel) - vráti počet zobrazení, ktoré potrebujeme na to, aby sme zrealizovali všetky objednávky zobrazovania reklamných boxov daného zadávateľa (3 body)
- **public** *String* topZadavatel() - vráti zadávateľa, ktorý má objednaný najväčší celkový počet zobrazení reklamných boxov (5 bodov)
- **public** *ReklamnyBox* zobrazBoxVKontexte(String slovo) - vráti zobrazovaný box, ktorý vo svojom texte obsahuje zadané slovo, a zvýši počet jeho zobrazení o 1. Ak takého boxu niet, vráti to, čo by vrátila metóda *zobrazBox*. Predpokladajte, že všetky slová v textoch sú oddelené iba medzerou. Ak je takých textov viac, vyberie ľubovoľný/náhodný z nich (7 bodov)
- + 10 bodov: vyriešte prípad, kedy parameter *slovo* obsahuje viacej slov a chcete vrátiť reklamný box, ktorého reklamný text obsahuje čo najviac z týchto slov.
- **public** *List*<*ReklamnyBox*> odstranNezobrazovane() - odstráni všetky už nezobrazované reklamné boxy a vráti referenciu na novovytvorený zoznam odstránených reklamných boxov (5 bodov)
- **public boolean** rozneAdresy() - vráti, či všetky reklamné boxy majú navzájom rôzne adresy, t.j. neexistujú 2 reklamné boxy, ktoré by smerovali na tú istú stránku (5 bodov)
- **public** *List*<*ReklamnyBox*> zobrazBoxy(int pocetBoxov) - vráti (nanajvyš) zadaný počet boxov na zobrazenie rovnakým spôsobom ako metóda *zobrazBox*. Pozor, nestačí však len *pocetBoxov* krát zavolať metódu *zobrazBox*. Vyžadujeme, aby vo vrátenom zozname boxov neboli žiadne 2 reklamné boxy smerujúce na tú istú webovú stránku, t.j. s rovnakou URL adresou stránky (5 bodov)

## Výnimky:

- Vytvorte nekontrolovanú výnimku `ZadavatelNeznamyException` a použite ju aspoň na dvoch vhodných miestach (4 body)

## Triedenie a komparátor (dokopy 5 bodov):

Vytvorte triedu *KomparatorBoxov* implementujúcu rozhranie `java.util.Comparator<ReklamnyBox>` s metódou:

- `public int compare(ReklamnyBox b1, ReklamnyBox b2)` – porovná boxy podľa úspešnosti – podielu kliknutí k počtu uskutočnených zobrazení boxu

V triede *Objednavky* implementujte inštančnú metódu:

- `public void zoradPodlaUspesnosti()` – usporiada reklamné boxy podľa úspešnosti