

Baigiamoji paskaita

OOP tolimesnių studijų kryptys
Egzamino nurodymai

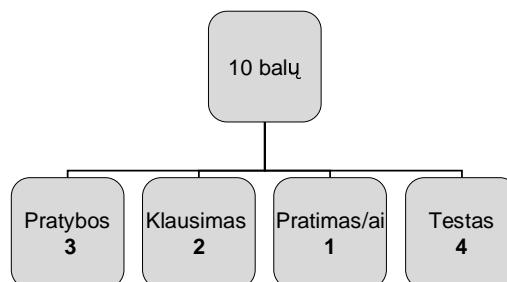
OO technologija

- Analizė
- Projektavimas
 - UML
 - Projektavimo šablonai (*design patterns*)
- Programavimas

OOP Java

- Java 1.5 versijos ypatybės
- Vidinės klasės (*Inner classes*)
- Refleksija
- Komponentės (*Beans*)
- Remote Method Invocation
- Java Native Interface
- Saugumas
- **GUI, JDBC, ...**

Rašto egzamino dalys



Pratybos : 3 balai

- Duomenis pateikia pratybų dėstytojas po paskutiniojo užsiėmimo
- vertinimas apvalinamas dešimtosiomis balo dalimis

Klausimas – 2 balai

- Klausimo - referato apimtis $\frac{1}{2}$ paskaitos temos
- Raštu pateikiamas išsamus atsakymas
- Būtinai pridedama (pseudo) kodo iliustracija
- Vertinimas diskretus – $\frac{1}{4}$ balo dalimis

Pavyzdys: *Klasės konstruktorius*

Pratimas/ai (1 balas)

Tai mini-užduotis/problema, kurios sprendimas - java klasė(ės) ar jos fragmentas (pseudo-kodas) ar/ir UML schema

1. Pratimas

Tegu duota klasė "Base". *Apibrėžkite išvestinę klasę*, pavadintą "Derived", kurioje perrašytas "String Object.toString()" metodas taip, kad grąžinama simboliu eilutė, skirtusi nuo bazinės klasės realizacijos grąžinamos eilutės priekyje ir gale pridėtu simboliu '*'.

(Pvz., jeigu bazinės klasės realizacija grąžintų eilutę "211", tai išvestinės klasės objektas - "**211**".)

2. Pratimas

Apibrėžkite klasę FOS2 išvestą iš FilteredOutputStream, perrašydami metodą **write(int b)** taip, kad kiekvienas į apibrėžtą srauto FOS2 objektą išvedamas baitas, kuris nėra tarpas "tikrajame" sraute būtų dubliuojamas (t.y., išvedamas 2 kartus).

3 pratimas

Daugiakampis susideda iš dvimačių Taškų, abiejų tipų objektai gali būti Stumdomi. Daugiakampiui galima skaičiuoti perimetrą. Turime atskirą Daugiakampio atvejį – Kvadratą, kuriam galima apskaičiuoti įstrižainės ilgį.

1. *Suprojektuokite* pabrauktąsias klases bei interfeisus UML diagrama (Progr. Sist. kursas)

2. *Realizuokite*.

Testo klausimai (4 balai)

- Klausimas ir keletas atsakymų, iš kurių tik vienas teisingas;
- Teisingas -geriausias, tiksliausias, labiausiai adekvatus klausimui atsakymas
- Gali būti reikalaujama trumpo komentaro
- Skiriamas balas: 0 arba "pilnas"
- Objektinė kalba: Java ver. 1.4

T1

Java klasę kompiliuojama komanda
(1) prepare (2) compile (3) javac (4) java.

Atsakymo NR.:

T2

Kuri iš šių reikšmių Javoje nėra primitivusis tipas (kitai tariant, yra objektas):

- (1) 'b' (2) "base" (3) 12.34 (4) true

Atsakymo NR.:

T3

Tegu kintamieji a, b, c yra to paties tipo ir rodo į skirtingus objektus. Įvykdžius šiuos sakinius

```
{ a = b; b = c; c = a; }
```

- (1) visi kintamieji rodytų į tą patį objektą;
(2) kintamieji rodytų į tris skirtingus objektus;
(3) kintamieji a ir c rodytų į tą patį objektą;
(4) bus sukurtos trys naujos objektų kopijos.

Atsakymo NR.:

T4

Ką galėtų reikšti sakinyš

```
a.f(100);
```

- (1) objektui a , išskviečiamas metodas f su parametru 100;
(2) sukuriamas masyvas vardu a , turintis 100 elementų;
(3) klasėje a apibrėžiamas kintamasis f , kurio reikšmė 100;
(4) objektui, į kurį rodo kintamasis f , kviečiamas šimtasis metodas.

Atsakymo NR.:

T5

Viešos (*public*) klasės A ir B priklauso tam pačiam paketui (*package*) vardu P . Tada teisinga:

- (1) klasė A gali naudoti privačius klasės B metodus
(2) klasės A ir B negali naudoti klasių vardų, priklausančių kitiems paketams
(3) kitų paketų klasės gali naudoti klases A ir B vardais $P.A$, $P.B$
(4) kitų paketų klasės negali kviesti klasių A ir B metodus.

Atsakymo NR.:

T6

Duotos klasės:

```
class Base {public int a = 1;}  
class Derived extends Base { public int a = 2;}
```

Ką išves šis kodo fragmentas:

```
{ Base b = new Derived();  
  System.out.println(b.a); }
```

- (1) 1
(2) 2
(3) įvyks RuntimeException

Atsakymo NR.:

T7

Metodų perkrova (*overloading*) reiškia:

- (1) tai kelių užduočių apjungimas į vieną metodą - objektinio programavimo esmė;
(2) tai užklojimo (*overriding*) sinonimas.
(3) metodų skaičius klasėje viršija leistiną ribą; (4) tai to paties vardo priskyrimas keliems panašioms metodams su besiskiriančia signatūra.

Atsakymo NR.:

T8

Tegu turime klases: C extends B {}, B extends A {}, o kintamieji a,b,c yra atitinkamų tipų (A,B,C) . Kuris priskyrimas sintaksiškai teisingas:

- (1) b = a; (2) c = a; (3) a = c; (4) b = new Object();

Atsakymo NR.:

T9

Jeigu klasė apibrėžta kaip *final* tai reiškia:

- (1) draudžiama apibrėžti išvestinę klasę;
- (2) sukūrus objektą - tos klasės egzempliorių,- draudžiama jį modifikuoti;
- (3) leidžiama apibrėžti išvestinę klasę bet draudžiama perrašyti bet kurį bazinės klasės metodą;
- (4) išvestinėje klasėje leidžiama perrašyti tik statinius elementus.

Atsakymo NR.:

T10

Jeigu klasės metodo parametro kintamojo vardas paslepia klasės (nestatinio) lauko vardą, metodo kode į šį lauką kreipsimės

- (1) naudodami `super` konstrukciją,
- (2) naudodami klasės vardą,
- (3) naudodami `this` išraišką,
- (4) kreiptis neimanoma;

Atsakymo NR.:

T11

Polimorfizmo principas objektiniame programavime reiškia:

- (1) galimybę apibrėžti klasių hierarchiją, bendriausias savybes nusakant bazinėje klasėje, o papildomus atributus išvestinėse klasėse
- (2) kintantį objektų sąveikos būdą, leidžiant objektams saugoti nuorodas į kitus objektus;
- (3) galimybę papildyti išvestinės klasės objektus naujais laukais, paliekant galioti ir bazinės klasės laukus;
- (4) skirtingą objektų, susijusių klasių hierarchijos ryšiu, reakciją į metodo iškvieta, priklausomai nuo objektų tipo.

Atsakymo NR.:

T12

Tegu klasė `BException` išvesta iš `AException`, o klasė `AException` išvesta iš `Exception` klasės. Kokia `catch` blokų try sakinyje tvarka yra sintaksiškai teisinga:

- (1) `Exception, AException, BException`,
- (2) `BException, AException, Exception`,
- (3) `AException, BException, Exception`.
- (4) Leistina bet kuri tvarka

Atsakymo NR.:

T13

Jeigu norime, kad metodas būtų prieinamas išvestinėse klasėse, tačiau nebūtų galima to metodo užkloti, jį apibrėšime kaip:

- (1) `protected abstract ...` (2) `protected final ...`
- (3) `private public ...` (4) `public abstract ...`

Atsakymo NR.:

T14

Ką išves ši programa (Derived.main()):

```
class Base{
    void f(){ System.out.print("a ");
    this.g(); }
    void g(){ System.out.print("b ");}
}
class Derived extends Base{
    void g() {System.out.print("c ");}
    void f() {System.out.print("d ");
    super.f(); }
    public static void main(String[] args) {
    Base b = new Derived(); b.f(); }
}
```

(1) ab (2) dac (3) dab (4) abcd

Atsakymo NR.:

T15

Kuris teiginys apie Iterator'ių **NE**teisingas:

1. Vienam kolekcijos objektui galima sukurti *kelis iteratorius*
2. Pasiekus kolekcijos pabaigą, iteruojama vėl *nuo pradžios*
3. Iteratoriaus interfeisas įgalina peržiūrėti *skirtingo tipo* kolekcijas
4. Atnaujinus kolekciją, iteratorius gali tapti *"neveikliu"*.

Atsakymo NR.:

T16

Kaip startuojama nauja gija:

1. Sukuriant **Thread** objektą,
2. Runnable objektui kviečiant metodą **run()**,
3. Pakartotinai kviečiant **main()** metodą,
4. Thread objektui kviečiant metodą **start** ?

Atsakymo NR.:

T17 (Progr. Sist.)

Langas vaizduoja objektus - paveikslėlius, kurie atsiunčiami iš nutolusio serverio. Kad pašymo metodas ilgai neužtruktų, buvo priimtas projektinis sprendimas: kol tikrasis paveikslėlis siunčiamas, pašymui pateikti "surogatinį" objektą. Koks projektavimo šablonas panaudotas:

- (1) Singleton'as (2) Iteratorius (3) Proxy
(4) Prototipas

Linksmų švenčių



Šaltinis: <http://www.kidsdomain.com/holiday/xmas/color.html>