

Kolokvijum se sastoji od dva dela:

- Prvi deo (30 poena) – sve potrebne klase se nalaze u okviru datog projekata. Potrebno ih je dovršiti na osnovu teksta/komentara u kodu. Takođe je dostupan i UML dijagram.
- Drugi deo (16 poena) – dati zahtevi obuhvataju pravljenje odgovarajućeg grafičkog okruženja.

## Prvi deo

Napisati na jeziku Java sledeće tipove:

### Paket geometrija

Apstraktan tip **Tacka** koji ima sledeće osobine:

- Javnu promenljivu **id** koja predstavlja jedinstveni broj tačke
- Javni podrazumevani konstruktor u kome se setuje vrednost **id** promenljive
- Javnu apstraktnu metodu **rastojanje** koja ima **realnu** povratnu vrednost. Metoda ima **jedan ulazni parameter** tipa **Tacka**. Metoda definiše mogućnost bacanja izuzetka tipa **TackeNisuUlstojDimenziji**.

Konkretni tip **TackeNisuUlstojDimenziji** koji predstavlja **grešku** i ima sledeće osobine:

- Private promenljive **t1** i **t2** tipa **Tacka**
- Javni konstruktor koji ima dva ulazna parametra tipa **Tacka** kojima se setuju vrednosti **t1** i **t2**
- Prepisanoj metodi **toString** koja vraća **poruku** da tačke **t1** i **t2** nisu u istoj dimenziji

Apstraktan tip **GeometrijskiOblik** koji ima sledeće osobine:

- Javnu apstraktnu metodu **dajPovrsinu** koja vraća realnu vrednost

### Paket geometrija.\_2D

Konkretni tip **Tacka2D** koji je **Tacka** i ima sledeće osobine:

- Privatne realne promenljive **x** i **y** koje predstavljaju 2D coordinate tačke.
- Javni konstruktor koji setuje **x** i **y** koordinatu tačke.
- Implementiran metod **rastojanje** iz klase **Tacka** koji računa rastojanje od tačke koja je prosleđena kao ulazni parameter
- Javni metod **rastojanje** koji ima **realnu** povratnu vrednost. Metod ima dva ulazna realna parametra **x** i **y** koji predstavljaju koordinate tačke u 2D prostoru od koje se računa rastojanje.
- Javne pristupne metode za sve promenljive

Apstraktan tip **GeometrijskaFigura** koja je **GeometrijskiOblik** i ima sledeće osobine:

- Javnu apstraktnu metodu **dajObim** koja vraća realnu vrednost

Konkretni tip **Pravougaonik** je **GeometrijskaFigura** i ima sledeće osobine:

- Promenljivu **gornjiLeviUgao** tipa **Tacka2D** koja ima **vidljivost samo u paketu** i predstavlja gornje levo teme pravougaonika
- Promenljivu **donjiDesniUgao** tipa **Tacka2D** koja ima **vidljivost samo u paketu** i predstavlja donje desno teme pravougaonika
- Javni podrazumevani konstruktor u kome se setuje **gornjiLeviUgao** na tačku (-1,1) i **donjiDesniUgao** na tačku (1,-1).
- Javni konstruktor koji prihvata **oba temena** i setuje ih. Ukoliko tačke nisu iz dvodimenzionalnog prostora baciti izuzetak **TackeNisuUlstojDimenziji**
- Implementiran metod **dajObim** koji računa obim pravougaonika
- Implementiran metod **dajPovrsinu** koji računa površinu pravougaonika

Konkretni tip **Trougao** je **GeometrijskaFigura** i ima sledeće osobine:

- Promenljive **a**, **b**, **c** koje su tipa **Tacka** i imaju vidljivost samo u paketu
- Javni konstruktor koji setuje sve tri tačke
- Implementiran metod **dajObim** koji računa obim trougla

- Implementiran metod **dajPovrsinu** koja računa površinu trougla
- Javne pristupne metode za sve promenljive
- Prepisan metod **toString** koja vraća string reprezentaciju svih tačaka trougla.

### Paket geometrija.\_3D

Konkretni tip **Tacka3D** koji je **Tacka2D** i ima sledeće osobine:

- Privatnu realnu promenljivu **z** koja predstavlja treću koordinatu tačke
- Javni konstruktor koji setuje **x**, **y** i **z** koordinatu tačke.
- Prepisan metod **rastojanje** iz klase **Tacka2D** koji računa rastojanje od tačke koja je prosleđena kao ulazni parameter. Ukoliko prosleđena tačka nije iz trodimenzionalnog prostora baciti izuzetak **TackeNisuUlstojDimenziji**
- Javni metod **rastojanje** koji ima **realnu** povratnu vrednost. Metod ima tri ulazna realna parametra **x**, **y** i **z** koji predstavljaju koordinate tačke u 3D prostoru od koje se računa rastojanje.
- Pristupnu metodu promenljivoj **z**.
- Prepisan metod **toString** koja vraća sve tri koordinate tačke.

Apstraktni tip **GeometrijskoTelo** koji ima sledeće osobine:

- Javnu apstraktnu metodu **dajZapreminu** koja vraća realnu vrednost

Apstraktni tip **ITeloSaBazom** koji ima sledeće osobine:

- Javnu apstraktnu metodu **dajPovrsinuBaze** koja vraća realnu vrednost
- Javnu apstraktnu metodu **dajObimBaze** koja vraća realnu vrednost
- Javnu apstraktnu metodu **dajBazu** koja vraća **GeometrijskiOblik**

Konkretni tip **Lopta** koji je **GeometrijskoTelo** i ima sledeće osobine:

- Promenljivu tipa **Tacka3D** koja se vidi samo u klasi i predstavlja centar lopte
- Realnu promenljivu **r** koja predstavlja poluprečnik lopte
- Javni konstruktor koji setuje obe promenljive
- Implementiran metod **dajZapreminu** koja računa zapreminu lopte
- Implementiran metod **dajPovrsinu** koja računa površinu lopte

Konkretni tip **TrostranaPiramida** koji je **GeometrijskoTelo** i ima osobine **tela sa bazom**:

- Promenljivu tipa **GeometrijskaFigura** koja predstavlja osnovu piramide i vidi se samo u klasi
- Promenljivu tipa **Tacka** koja predstavlja vrh piramide
- Javni konstruktor koji prima tri tacke trougla u osnovi i tacku koja predstavlja vrh piramide i setuje obe svoje promenljive
- Implementiran metod **dajZapreminu** koja računa zapreminu trostrane piramide
- Implementiran metod **dajPovrsinu** koja računa površinu trostrane piramide
- Implementiran metod **dajPovrsinuBaze** koja računa površinu trougla koji je u osnovi piramide
- Implementiran metod **dajObimBaze** koji računa obim trougla koji je u osnovi piramide
- Implementiran metod **dajBazu** koji vraća osnovu trostrane piramide

### Podrazumevani paket

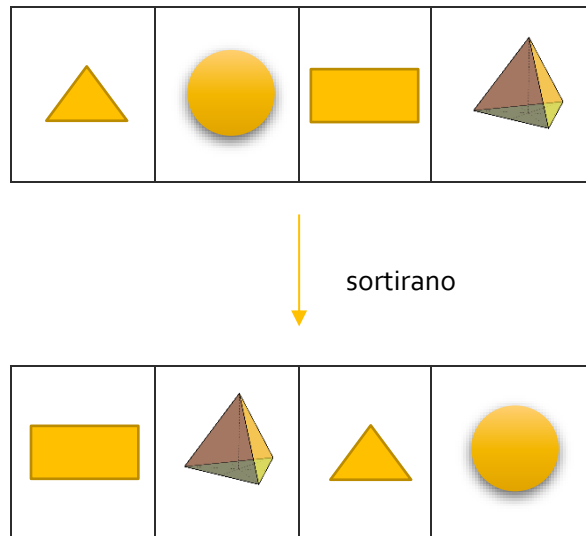
Konkretni tip **Prostor** koji ima sledeće osobine

- Promenljivu koja predstavlja niz geometrijskih oblika
- Javni konstruktor kome se prosleđuje kapacitet niza geometrijskih oblika
- Javni metod **dodajGeometrijskiOblik** koji prihvata jednu promenljivu tipa **GeometrijskiOblik** i skladišti je u niz
- Javni metod **dajZapreminuSvihOblikaKojiJelmaju** koja vraća ukupnu zapreminu svih geometrijskih oblika koji imaju zapreminu
- Javni metod **dajPovrsinuSvihOblika** koja vraća ukupnu površinu svih geometrijskih oblika
- Javni metod **dajBazeSvihOblikaKojiJelmaju** koji vraća niz baza svih oblika koji ih imaju
- Javni metod **sortirajGeometrijskeOblike** koji sortira tela prema zapremini, a figure prema obimu - **bonus (2 poena)**

Testnu klasu **Test** u kojoj je potrebno:

- Kreirati jedan **Prostor** koji može imati najviše 5 geometrijskih oblika

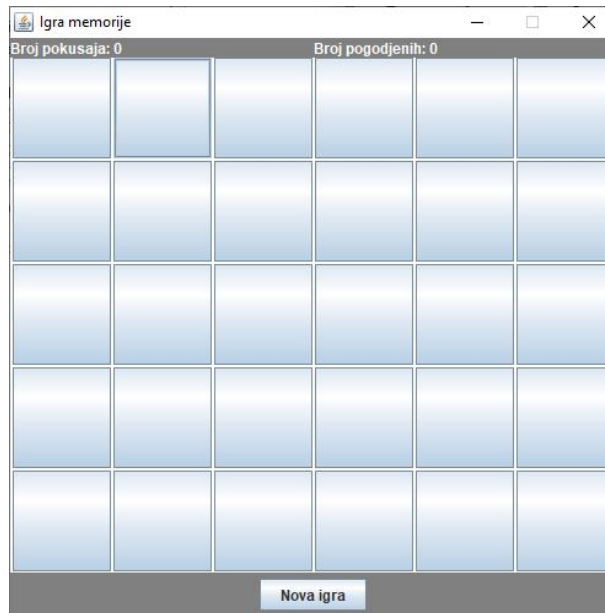
- Kreirati trougao čije tačke imaju koordinate (2, 5), (1, 8) i (4, 7)
  - Dodati kreirani trougao u prostor
  - Kreirati loptu čiji je centar tačka (4, 5, 9) i poluprečnik 10. Takođe je dodati u prodor.
  - Kreirati pravougaonik čije je gornje levo teme tačka (4, 4) a donje desno teme tačka (5, 2). Dodati ga u prostor.
  - Kreirati piramidu čija osnova ima tačke (2, 4, 8), (5, 3, 8) i (2, 2, 8), dok je vrh piramide tačka (3, 3, 15). Dodati je u prostor.
- Ispisati:
- po jedan podatak o svim geometrijskim oblicima prostora, ali tako da se za geometrijska tela štampa njihova zapremina, a za geometrijske figure njihov obim.
  - ukupnu površinu svih geometrijskih oblika
  - ukupnu zapreminu geometrijskih oblika
  - povrsine svih baza geometrijskih tela
  - sortira prostor tako da se tela sortiraju prema zapremini, a figure prema obimu i ispisati na isti način kao pod a)



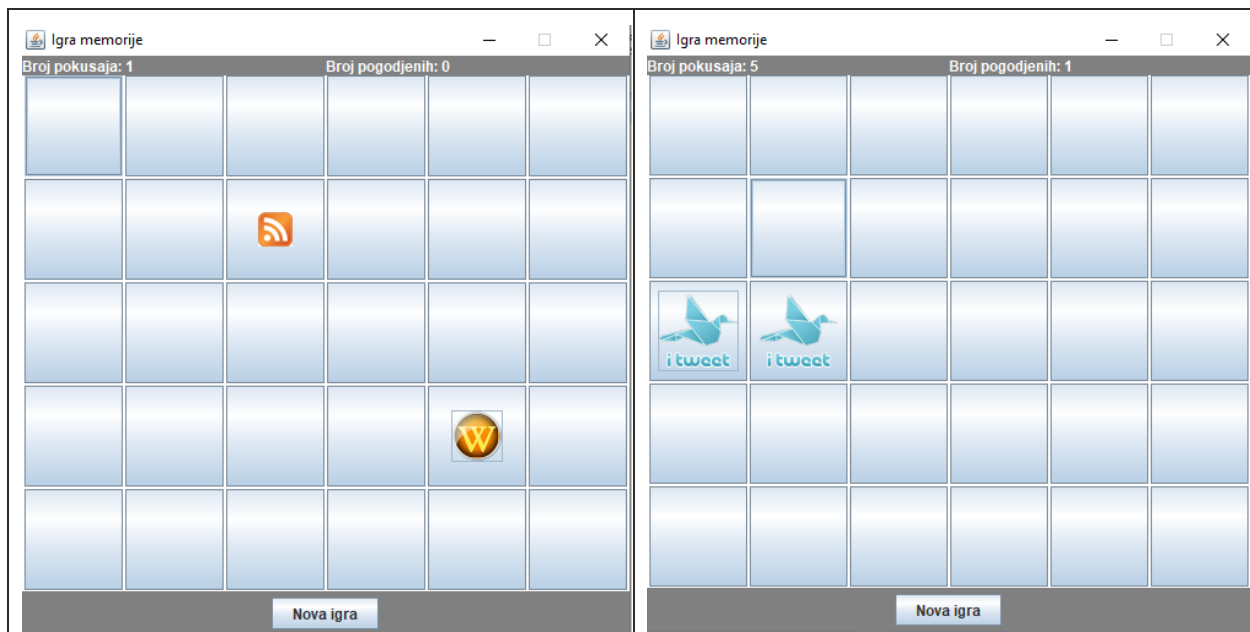
	Trougao	Pravougaonik	Lopta	Trostrana piramida
Zapremina	-	-	$V = \frac{4}{3} r^3 \pi$	$V = \frac{B \cdot H}{3}$ gde je B površina baze.
Povrsina	$P = \sqrt{s(s-a)(s-b)(s-c)}$ $S = \frac{a+b+c}{2}$	$P = ab$	$P = 4r^2 \pi$	$P = B + O$ gde je B površina baze, O površina omotača.
Obim	$O = a + b + c$	$O = 2a + 2b$		

## Drugi deo

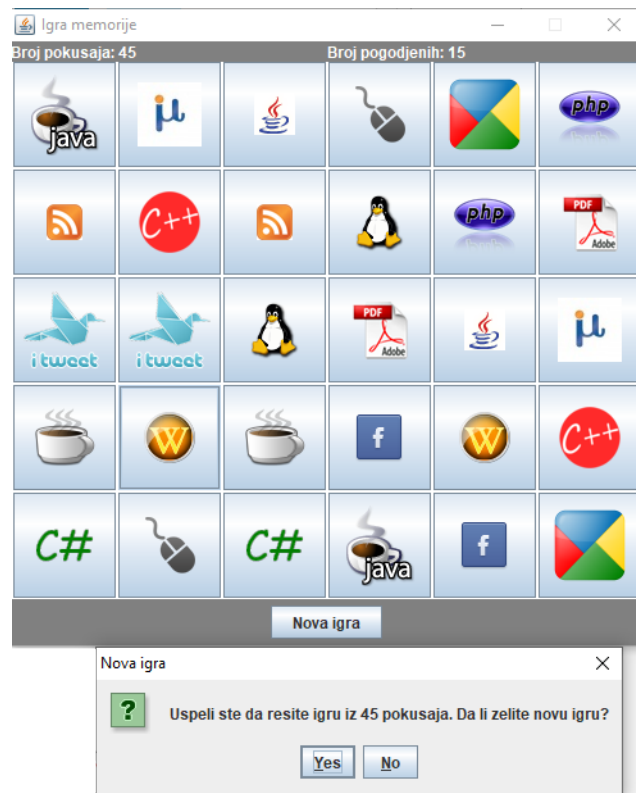
Napisati program koji implementira logiku igre Memorije. Grafički prikaz igre se može videti na slici.



Igra se na taj način što se biraju dva polja. Ukoliko klikom na data polja, se otvore različite sličice, one se automatski zatvaraju novim izborom dva polja. Ukoliko su sličice da poljima iste, polja ostaju otvorena.



Igra se završava kada se otvore sva polja.



Klikom na dugme **Nova igra** ona se inicijalizuje na početno stanje.