

Java Тест

Задача 1: Да се въведат 10 числа от клавиатурата и да се изведат най-малкото и най-голямото измежду тези числа.

Задача 2: Да се изведат всички нечетни числа от 1 до 100 в обратен ред.

Задача 3: Въведете 5 числа от клавиатурата. Определете кои от тях са прости (делят се само на 1 и на себе си) и изведете тяхната сума (само на простите числа).

Задача 4: Да се въведат низов стринг и буква от клавиатурата и да се преброи колко пъти буквата е срещана в стринга. Резултатът да се изведе на екрана.

```
string s = scanner.next();  
s.charAt(0); + for
```

Задача 5: Да се въведе число от клавиатурата, по-голямо от 3, и да се създаде нов масив с големина равна на това число. Намерете и изведете на екрана първите 3 най-малки числа от масива.

Задача 6. Създайте интерфейс Bounceable с два метода – void bounce() и void roll(). Създайте абстрактен клас Ball който имплементира този интерфейс, и дефинира полетата double bounceFactor и double radius. Те трябва да могат да бъдат наследявани от класове наследници. Създайте конструктор по подразбиране и конструктор с аргументи за полетата на класа. Създайте и публични методи за достъп – get- и set- методи. Предефинирайте метода toString().

Създайте класове Baseball, Football и Volleyball, които наследяват класа Ball и предефинират методите bounce() и roll() – извеждайте просто съобщение на екрана каква тип топка извърша bounce или roll. Създайте и необходимите конструктори.

is bouncing

Създайте клас Playground с main метод и създайте няколко инстанции на топки. Демонстрирайте използването на методите bounce и roll.

Задача 7. Създайте клас Student с полета за име, години, факултетен номер и успех. Създайте необходимите конструктори и get- и set- методи. Предефинирайте toString() и equals(). Създайте нов клас MathClass в който има поле List<Student> students. Инициализирайте списъкът с празен списък в конструктора на MathClass. Създайте в MathClass:

- Метод за добавяне на нов студент в класа
- Метод за намиране на студент по факултетен номер. (Методът трябва да връща обект от тип Student)
- Метод за намиране на студента(студентите) с най-нисък успех.

equals

metoda trqbwa da vrsta spisak

Методът трябва да връща списък от студенти, тъй като може повече от един студент да са с най-нисък успех **comparator**

- Метод който принтира в конзолата студентите, сортирани по успех в намаляващ ред (от най-висок към най-нисък успех). Принтирайте имената на студентите и техния успех. Забележка: Използвайте

Comparator<Student> по подобие на примерните задачи, качени в лекция 7.

- Метод, който приема обект от тип студент като аргумент, намира го в списъка и го изтрива. **iterator**

Задача 8: Напишете програма, която чете текст от файл, обръща съдържанието и го записва в друг файл. Пътищата до файловете могат да бъдат зададени директно в кода.