

1. **Balancing brackets:** აირჩიეთ შესაბამისი `data structure` და დაწერეთ პროგრამა, რომელიც შეამოწმებს არის თუ არა გადმოცემულ `string`-ში სწორად დაბალანსებული ფრჩხილები. ეს `string` შეიცავს მხოლოდ შემდეგ სიმბოლოებს: `()`, `{}`, და `[]`. დაბალანსებულის მაგალითია: `(){}([])` ხოლო დაუბალანსებელია `{([])}`
2. **Point Calculations:** აიღეთ 2D ან 3D სიბრტყე, სიბრტყეზე წერტილი წარმოადგინეთ `Tuple`-ის სახით და დაწერეთ მეთოდი რომელიც მიიღებს ორ ასეთ წერტილს და გამოთვლის მანძილს ამ წერტილებს შორის.
3. **Multiple return values:** გამოიყენეთ აქამდე დაწერილი მეთოდი, სადაც `out` პარამეტრით აბრუნებდით მნიშვნელობას და გადაწერეთ ის `Tuple`-ების გამოყენებით.
4. **Custom sorting:** შექმენით წიგნის კლასი, რომელსაც ექნება შემდეგი `property`-ები: ავტორის სრული სახელი, წიგნის სახელი, გამოშვების წელი, ISBN, ჟანრი (`enum`). შექმენით წიგნების `List` (უბრალოდ `List<Book>` შექმენით `built-in List`-ის გამოყენებით) შემდეგ დაწერეთ რამდენიმე `IComparer<Book>`-ის იმპლემენტაცია, რომელიც საშუალებას მოგცემთ წიგნების სია დაალაგოთ სხვადასხვა ველის მიხედვით (ველების არჩევა თქვენთვის მოგვინდია)