

Implementoi oma pinoluokka, käyttäen listaa tietorakenteena.

- Toteutuskieli C#
- Osaa käsitellä vain kokonaislukuja
- Luokan nimi IntStack
- Julkiset metodit
 - Push(int item), lisää item-argumentin pinon päällimmäiseksi, ei paluuarvoa
 - Pop(), poistaa ja palauttaa pinon päällimmäisen jäsenen. Jos pino on tyhjä, heittää virheen `InvalidOperationException` viestillä "Stack empty"
 - Peek(), palauttaa pinon päällimmäisen jäsenen poistamatta sitä. Jos pino on tyhjä, heittää virheen `InvalidOperationException` viestillä "Stack empty"
 - Clear(), tyhjentää pinon, ei paluuarvoa
 - Contains(int item), tarkistaa sisältääkö pinon item-argumentin arvon. Löytyessä palauttaa true, jos ei löydy palautuu false
- Ominaisuus Count, joka kertoo jonon sisältämien jäsenten määrän
 - Voidaan lukea ulkopuolelta, kirjoitus vain luokan sisällä
- Konstruktorimetodilla ei ole argumentteja
- Ei tarvitse pohtia algoritmien suorituskkyä
- Luokan ja metodien nimet (ja argumenttien tyypit) pitävät olla määritysten mukaisia
 - Poikkeaminen näistä johtaa palautuksen hylkäämiseen (koska luokkasi testataan ohjelmallisesti ja opettaja ei tee siihen mitään muutoksia)
- Palautukseen liitetään vain luokan implementoiva (toteuttava) koodi, ei yhtään mitään muuta
 - Eli alkaa `class IntStack {` ja loppuu `}`
- Koodi ei saa tulostaa mitään ylimääräisiä
 - Muista siis poistaa kaikki mahdolliset tarpeettomat tulostuskutsut
- Määriteltujen julkisten metodien lisäksi luokassasi voi olla tarpeellisia yksityisiä metodeja ja muuttujia
- Sinun pitänee testailla luokkaa huomattavan monella tavalla, jotta voit varmistua sen toimivuudesta