1. Feladat: Kártyajáték szimulátor

Készíts egy programot, amely szimulál egy egyszerű kártyajátékot!

* + Kérj be egy játékos nevét.
  + Kérj be egy kártyalapot: "piros" vagy "fekete".
  + Generálj véletlenszerűen egy kártyalapot.
  + Ellenőrizd, hogy a játékos nyert-e a kártyajátékon.
  + Ha a játékos eltalálta, írd ki: "Gratulálok, nyertél!"
  + Ellenkező esetben írd ki: "Sajnálom, vesztettél."

1. Feladat: Országok és fővárosok

Készíts egy programot, amely kezeli az országok és fővárosok adatait!

* + Hozz létre egy listát, amely tárolja néhány országot és annak fővárosát. pl (Magyarország-Budapest)
  + Írj metódust, mely bekér egy országot a felhasználótól.
  + Írj metódust, mely ellenőri, hogy az ország szerepel-e a listában.
  + Ha igen, írd ki a fővárost.
  + Ha nem, írd ki: "Az adott ország nem található a listában."

1. Feladat: Egyetemista hallgatók kezelése

Készíts egy programot egyetemista hallgatók kezelésére szolgál!

* + Hozz létre egy **Hallgato** osztályt, amely reprezentál egy egyetemista hallgatót. Minden hallgatónak legyen neve, születési éve és szakja.
  + Hozz létre egy **listát, amely** hallgatókat tárolja. Olvasd be az egyetemistak.txt fájlt és tárold az adatokat a listában.
  + Írj metódusokat a hallgatók hozzáadásához, eltávolításához és listázásához.
  + Kérj be egy egyetemista nevét a felhasználótól. Ellenőrizd, hogy az adott név szerepel-e az egyetemisták között.
  + Ha igen, kérj be új adatokat (születési év és szak) a felhasználótól, majd módosítsd az adott egyetemista adatait.
  + Ha nem, írd ki: "Az adott névvel rendelkező egyetemista nem található a listában."
  + Számold ki és írd ki, hány olyan egyetemista van, aki az 1990-es években született.
  + Határozd meg, melyik a legnépszerűbb szak a hallgatók körében.
  + Számold ki és írd ki az átlagos életkort az összes egyetemista számára.