* Room
  + splitTo(*r2*: Room): void
    - csak azért jelöltem *r2*-vel, hogy könnyebb legyen magyarázni
    - ha meghívódik, átadja r2-nek a szomszédjainak, és az itemjeinek, a felét.
    - Így valósul meg az osztódás
  + mergeTo(*r2*: Room): void
    - azért jelöltem *r2*-vel, hogy könnyebben legyen magyarázni
    - ha meghívódik, odaadja az összes szomszédját, és itemjét az *r2*-nek
    - Így valósul meg az egyesülés
      * miután átadtuk a cuccokat r2-nek, törlődik a szoba
  + copyToRoom(*r2:* Room): void
    - *r2* jelölés same
    - ha meghívódik, átmásolja a szomszédjati, itemeit, és a benne lévő charactereket r2-be
    - A camembert letételénél használom
* Student
  + A transistor változó lehetne lista (pl transistorList néven)
    - Azaz a player által felvett tranzisztorokat egy külön listában tárolhatnánk, úgy talán könnyebb kezelni őket
      * Potyók Csábor approved
    - Annyi, hogy amikor az inventoryt nézzük, hogy mennyi hely van még, akkor bele kell számolni ezt is
      * ez azt jelentené tippre, hogy kódban nem 1 sor, hanem 2
    - Megj: GUI-ban akár lehetnek külön is szedve
  + addTransistor(Transistor): void
    - beletesz a student transistor listájába a paraméterül kapott elemet
  + getTransistor(*id*: int): Transistor
    - a transitorList listának visszaadja az *id-*aidik elemét
    - Ez a transistor párosodásnál jön elő