

Během zkoušky aktivně komunikujte se zkoušejícími, nebojte se zeptat. Nad zadáním přemýšlejte, často je těžší zjistit *co* řešit než *jak*. Kód pište čistě a průběžně ho vylepšujte. Nesnažte se vyřešit všechny příklady naráz — hodnotit budeme především podle kvality vašich myšlenek a vašeho kódu.

Při implementaci iterátorů nepište nepovinnou metodu `remove`.

1. Napište adaptéry, které z instancí tříd `Student` a `Course` vytvoří dvojice. Nezapomeňte, že dvě n-tice jsou stejné, pokud mají stejné složky. Kód duplikujte co nejméně.
2. Pomocí spojového seznamu naimplementujte kód třídy `Set`.
3. Do třídy `Set` přidejte podporu pro události.
4. Přepište implementaci třídy `Set` tak, aby využívala návrhový vzor nulový (prázdný) objekt. Podmínky redukujte na absolutní minimum, ideálně jen na test, jestli se dva objekty rovnají (ne na porovnání s `null`).
5. Naimplementujte metodu `Set.filter`, která vrátí `SetView` s vyfiltrovanými n-ticemi.
6. Pomocí dynamické proxy naimplementujte adaptér, který dostane objekt, tři jména jeho atributů a vyrobí z něj trojici.