

## Cvičení se spojovými seznamy celých čísel

Následující úlohy řešte pro jedno- i obousměrně zřetěžené seznamy a je-li to možné, také pro seznamy uspořádané podle velikosti klíče.

Zkontrolujte svá řešení pomocí nevelkého (10-20) souboru náhodně generovaných hodnot, výsledky vypisujte na obrazovku. Preferujte efektivní a paměť nezatěžující postupy, při změně struktury seznamu pokud možno manipulujte jen s ukazateli a ne s hodnotami prvků.

Porovnejte efektivitu vytvoření dlouhého vámi implementovaného seznamu vůči rychlosti vytvoření seznamu obsahujícího stejnou informaci pomocí knihovní třídy, např. `LinkedList` (v Javě). Porovnejte i další operace se seznamy.

1. Vytvořte třídu `P` reprezentující prvek seznamu a třídu `S` reprezentující seznam. Třída `S` bude obsahovat tyto metody
  - přidání prvku s danou hodnotou na začátek seznamu,
  - odebrání prvního prvku (pouze z neprázdného seznamu!),
  - výpis prvků seznamu,
  - test neprázdnosti seznamu.
2. Vytvořte potomka `S2` třídy `S`, který bude navíc disponovat metodami
  - navrácení počtu prvků v seznamu (v konstantním čase),
  - přidání prvku na konec seznamu,
  - odebrání posledního prvku ze seznamu,
  - připojení seznamu na konec jiného seznamu,
  - vytvoření duplikátu seznamu,
  - porovnání dvou seznamů zda obsahují tutéž posloupnost hodnot,
  - zjištění, zda je seznam uspořádan v nerostoucím, neklesajícím pořadí (vč. identity všech hodnot) nebo zda není uspořádan (vše v jediném průchodu).
3. Vytvořte potomka `S3` třídy `S2`, který bude navíc disponovat metodami
  - nalezení prvního prvku seznamu se zadanou hodnotou a navrácení jeho pořadového čísla v seznamu (-1, pokud neexistuje),
  - vložení nového prvku se zadanou hodnotou těsně za první prvek seznamu s další zadanou hodnotou (např. seznam: 2,8,6,5, 9; `vlozZaCo(6, 11)`; seznam: 2,8,6,11,5,9) pokud druhá zadaná hodnota v seznamu není, přidá se nový prvek na konec,
  - odstranění ze seznamu prvního prvku se zadanou hodnotou (zajímavé v jednosměrném seznamu!),
  - inverzi seznamu, tj. obrácení pořadí prvků v seznamu.
4. Vytvořte potomka `S4` třídy `S3`, který bude navíc disponovat metodami
  - rozdělení seznamu na dva nové seznamy, kde první bude obsahovat všechny prvky původního seznamu s hodnotou menší než zadaný parametr a druhý seznam bude obsahovat všechny ostatní prvky, v obou nových seznamech budou prvky uloženy ve stejném pořadí jako v původním seznamu,
  - implementující libovolné další nápady.