

## Domácí úkol číslo 2: Seznamy

### A. Generování seznamů

1. vygenerujte rostoucí posloupnost N lichých čísel a uložte ji do seznamu.
2. vygenerujte klesající posloupnost N lichých čísel a uložte ji do seznamu.
3. vygenerujte prvních N prvočísel a uložte je do seznamu.
4. vygenerujte posloupnost  $N!$  (1,2,6,24,120,...) a uložte ji do seznamu.
5. vygenerujte N-tou řádku Pascalova trojúhelníka.

### B. Seznamy

1. Nalezněte všechny duplicity v seznamu, do výsledku dejte každý duplicitní prvek pouze jednou. `?-dupl([a,b,1,c,3,d,2,2,f,3,2],X)`.  $X=[3,2]$
2. Smažte všechny unikátní prvky ze seznamu. `?pouze_dupl([6,2,3,3,5,2,3,1,4],X)`.  $X=[2,3,3,2,3]$
3. Smažte všechny duplicity v seznamu. `?unikatni([a,b,c,d,a,b,e,f],X)`.  $X=[a,b,c,d,e,f]$ .
4. Spočítejte počet výskytů prvku v seznamu. `?-vyskyty([a,b,c,1,2,a,b,1,1,1],X)`.  $X=[[a,2],[b,2],[c,1],[1,4],[2,1]]$
5. Máme zadán seznam, který reprezentuje HashMapu. Naprogramujte predikat `hashmapADD`, který přidá dvojici (klíč, hodnota) do HashMapy tak, že pokud klíč v mapě není vytvoří novou dvojici, jinak opraví stávající hodnotu u daného klíče.  
`?-hashmapADD([[a,1],[b,14],[c,30]], d, 12, X)`.  $X=[[a,1],[b,14],[c,30],[d,12]]$   
`?-hashmapADD([[a,1],[b,14],[c,30]], b, 12, X)`.  $X=[[a,1],[b,12],[c,30]]$