



Demonstrature 3

Zadatak 1. Implementirajte funkciju koja prima nepraznu konačnu listu te vraća uređeni par (a, b) , gdje je a prvi element liste, a b zadnji element liste.

Zadatak 2. Implementirajte funkciju koja prima listu x te vraća listu parova uzastopnih elemenata od x . Npr. ako lista sadrži elemente $[a, b, c, d]$, funkcija treba vratiti $[(a, b), (b, c), (c, d)]$.

Zadatak 3. Implementirajte funkciju koja prima listu tipa `Double`, a vraća listu korijenjenih elemenata s parnim indeksima. Pretpostavite da indeksi kreću od 0.

Zadatak 4. Implementirajte funkciju koja prima string koji se sastoji isključivo od malih slova engleske abecede i razmaka te iz njega izbacuje sve samoglasnike. Koristite pattern matching.

Zadatak 5. Implementirajte funkciju *binom* koja prima redak iz Pascalovog trokuta reprezentiran listom cijelih brojeva, a vraća sljedeći redak.

Zadatak 6. (Zadatak s drugih vježbi) Implementirajte funkciju *apply* koja:

- prima funkciju f , element a proizvoljnog tipa i prirodan broj n (ili nulu);
- n puta primjenjuje funkciju f na a ;
- vraća rezultat n -te primjene

Npr. za $n = 3$ funkcija *apply* treba vraćati $f(f(f(a)))$.

Zadatak 7. Koristeći funkcije *binom* i *apply* iz prethodnih dvaju zadataka, implementirajte funkciju *pascal* koja prima $n \in \mathbb{N}_0$ te računa n -ti redak Pascalovog trokuta. Pretpostavite da je nulti redak Pascalovog trokuta jednak $[1]$.