Úlohy na cvičenia pre 3. týždeň semestra

1. Na rozohriatie: **Hod mincou**

Budeme hádzať mincou pomocou príkazu ***rbinom()*.**Testujte postupne situáciu pri 10 hodoch, pri 100, 1000, 10 000, 100 000 hodoch, aká je pravdepodobnosť, že hodíte „hlavu“.  
Fyzické sčítanie pozitívnych výsledkov realizujte minimálne 5-timi rôznymi spôsobmi (aspoň jeden cyklus, aspoň dva spôsoby cez funkčné programovanie).[[1]](#footnote-1)

Sledujte aká časová a pamätová náročnosť pri každom spôsobe, ktorý ste naprogramovali je potrebná. Sledujte ako sa jednotlivé programátorské štýly správajú v tomto a aj v nasledujúcich algortimoch. Sledujte, kde sú ich slabé a silné stránky.

1. **„Priemer“ kladných hodnôt**

Vytvorte vektor náhodne generovaných **celých čísel z intervalu 0 až 10 s normálnym rozdelením** (napr. ***rnorm()*** )– opäť postupne s dĺžkou 100 – 1 000 000 čísel. Vypočítajte aritmetický, geometrický, harmonický a kvadratický priemer z týchto čísel. Vymyslite aspoň 5 spôsobov, ako túto úlohu zrealizovať (napr. ***mean()***, cyklus, funkčné programovanie) pre každý z priemerov. Najmä pri väčšom dátovom objeme sledujte, či je z časového hľadiska lepšie vygenerovať celú množinu, uložiť ju, alebo ju generovať a počítať po dávkach – samozrejme s ohľadom na použitý algoritmus. Identifikujte slabé a silné stránky všetkých postupov.[[2]](#footnote-2)

**Aritmetický** **priemer**- ,

**Geometrický** **priemer** 

Je zrejmé, že geometrický priemer má zmysel iba pre dáta, v ktorých sú všetky hodnoty kladné čísla. Geometrický priemer sa na rozdiel od aritmetického priemeru používa na koeficienty, napr. na výpočet priemerného rastu: ak rast cien bol postupne 20 %, 10 %, potom 15 % pokles a 10 % rast, tak priemerný rast sa rovná ( čiže priemerný rast je približne 5,4 %. Toto číslo vyjadruje, že výsledná cena by bola taká istá aj v prípade, ak by rast bol konštantný, každý rok 5,4 %

Asi to veľmi rýchlo zistíte, že väčšej dátovej množine rýchlo narazíte na limity aritmetických operácií, takže jeden z možných nápadov pre veľké dátové množiny - pri použití logaritmov možno súčiny zmeniť na súčty a umocňovanie na súčin, 

**Harmonický priemer **

Harmonický priemer sa používa na určenie stredu variability takého znaku, ktorý cez určitú

konštantnú hodnotu je v nepriamom vzťahu s iným znakom. Súčet takýchto hodnôt znaku nedáva logický zmysel. Harmonický priemer sa používa na charakterizovanie hodnôt, ktoré predstavujú napríklad výkonové limity – teda dosiahnuť u každej osoby ten istý výkon pri rôznom čase alebo rôzny výkon za jednotku času (1. osoba urobí prácu za hod, teda jej hodinový výkon je, …,atď.) V prípade rôznych vzdialeností a rovnakých časov sa však musí použiť aritmetický priemer.

**Kvadratický** **priemer**  . Kvadratický priemer sa obyčajne používa vo fyzike, kde sa často označuje ako efektívna hodnota.

1. Zopakujte úlohu 2 **pre vektor náhodne generovaných reálnych čísel z intervalu .** Kedy a v ktorých prípadoch budete pozorovať výraznú zmenu oproti predchádzajúcemu prípadu (spotreba pamäte, čas potrebný na výpočet...) Ako overíte, či sa sa na vypočítaný výsledok môžete spoľahnúť?
2. **RMSD – root mean square deviation s podmienkou**

Opäť vytvorte **vektor náhodne generovaných reálnych čísel z intervalu  s normálnym rozdelením** – opäť postupne s dĺžkou 100 – 1 000 000 čísel. ***Naučte sa v tomto príklade používať funkcionálnu podmienku!*** To znamená, že vymyslíte aspoň 5 spôsobom (jeden cyklus, aspoň dva rôzne funkcionálne prístupy) ako z vami vygenerovanej dátovej množiny vyberiete len kladné čísla (len záporné čísla, len čísla z nejakého intervalu ...) a pre tento výber vypočítate aká bude RMSD

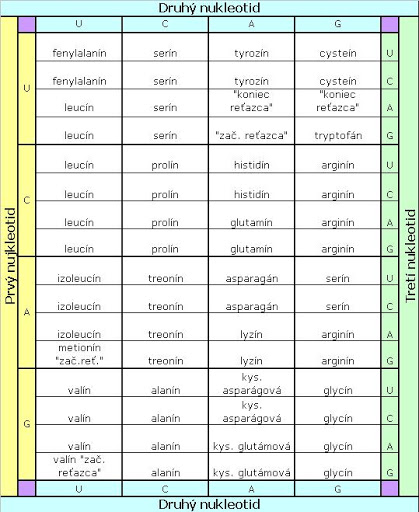


Pre každý prístup (každý naprogramovaný spôsob) a každú dĺžku vektora testujte aj pamätové aj časové nároky a overte hraničné limity použitia jednotlivých programátorských štýlov.

1. **Trochu genomiky - (práca s typom char)**

Stiahnite si z drivu kompletnú genetickú informáciu pre covid 19 (NCBI Reference Sequence: NC\_045512.2)[[3]](#footnote-3) a urobte jeho analýzu pre jednoduché základné úlohy. Pre každú úlohu (ako obvykle) aspoň 5 rôznych spôsobov ako sa to dá naprogramovať a minimálne dva (tri?) funkcionálne

* Zistite, koľko obsahuje písmeno A, C, T, G.
* Zistite koľko krát sa v reťazci vyskytujú všetky základné aminokyseliny (Jednotlivé dusikaté bázy – písmená – navzájom utvárajú trojice. Každá trojica predstavuje kodón – jednu aminokyselinu. Spájaním kodónov, aminokyselín, vznikajú kódy - gény - pre funkčné bielkoviny.



1. Pre úplnosť – termín „rôzny spôsob“ sa chápe ako myšlienkovo úplne iný prístup ku danému problému. Do tejto kategórie nespadá postup, že použijem inú knižnicu a mierne iné príkazy na zrealizovanie toho istého zadania. Chápe sa tým procedurálne programovanie, objektové programovanie, funkcionállny prístup, conditional functions, pure functions, anonymous functions .... [↑](#footnote-ref-1)
2. Zdroj http://math.ku.sk/data/portal/data/zbornik2007/Articles/Kulcar\_Ladislav.pdf [↑](#footnote-ref-2)
3. Ak vás zaujíma iná (nie základná mutácia, stiahnite si dáta z databázy <https://www.ncbi.nlm.nih.gov/labs/virus/vssi/#/virus?SeqType_s=Nucleotide&VirusLineage_ss=SARS-CoV-2,%20taxid:2697049> , klik na meno, potom vľavo hore formát FASTA a vpravo hore sent to file [↑](#footnote-ref-3)