

## Uvod u Bioinformatiku, Julski rok

Industrija 4.0

Školska godina 2022/2023

### Uputstvo:

Na Desktop-uu se nalazi folder sa nazivom BIO.JUL u kome se nalaze nepotpuni kodovi rešenja zadataka koje treba dopuniti kako bi traženi algoritmi bili implementirani. Folder preimenovati u format bio\_ispitniRok\_Prezime\_Ime\_BrIndeksa (npr. bio\_sept1\_Peric\_Pera\_4005-2021) i sve kodove čuvati u preimenovanom folderu dopunjujući nepotpune kodove.

**NAPOMENA: Kodovi koji se ne kompajliraju (imaju sintaksne greške) automatski nose 0 poena.**

1. [12.5p] Napisati kod *Faster Frequent Words* algoritma koji na efikasan način pronalazi skup najzastupljenijih podniski dužine  $k$  u datoj sekvenci *text*.

Primer:

```
text = 'CATAAATTTTCGTATGTATCAAAATTTTGTACTATCACATAAATTTTCGTATGTATCAAAATTTTGTACTATCA'
```

```
k = 6
```

```
Izlaz: ['AAATTT']
```

2. [12.5p] Napisati kod algoritma *Leaderboard Cyclopeptide Sequencing* koji pronalazi (ciklo)peptid čiji je ciklični spektar najsaglasniji sa datim spektrom *target\_spectrum*, pri čemu se prilikom pretrage u svakoj iteraciji lista peptida-kandidata krati sa granicom za odsecanje  $N$ .

Primer:

```
target_spectrum = [0, 99, 113, 114, 128, 227, 257, 299, 355, 356, 370, 371, 484]
```

```
N = 10
```

```
Izlaz: 'QNLE'
```

3. [12.5p] Napisati kod algoritma *Needleman-Wunsch* koji pronalazi optimalno globalno poravnanje za date dve sekvence, pri čemu je skor uparivanja istih karaktera jednak 2, skor uparivanja različitih karaktera -1, a poravnanje karaktera sa prazninom -2.

Primer:

```
seq1 = 'AGTACGCA'
```

```
seq2 = 'TATGC'
```

```
Izlaz: Score: 1
      Alignment:
      AGTACGCA
      --TATGC--
```

4. [12.5p] Napisati kod *UPGMA* algoritma koji konstruiše ultrametrično evolutivno stablo za matricu rastojanja  $D$  dimenzije  $n \times n$ .

Primer:

```
D = [[0, 3, 4, 3],
      [3, 0, 4, 5],
      [4, 4, 0, 2],
      [3, 5, 2, 0]]
```

```
n = 4
```

```
Izlaz: Root = [0, 3, 1, 2]: 9.333333333333334
      0
      Neighbors: []
      1
      Neighbors: []
      2
      Neighbors: []
      3
      Neighbors: []
      [1, 2]:6.0
      Neighbors: [(' [1]:0', 6.0), (' [2]:0', 6.0)]
      [3, 1, 2]:6.5
      Neighbors: [(' [3]:0', 6.5), (' [1, 2]:6.0', 0.5)]
      [0, 3, 1, 2]:9.333333333333334
      Neighbors: [(' [0]:0', 9.333333333333334), (' [3, 1, 2]:6.5', 2.8333333333333334)]
```