AvB – DN3

Naloga 1. – Poravnava več zaporedij

Nagrada 0 za ujemanje ter kazen 1 za zamenjavo/delecijo/insercijo. Za razdaljo upoštevamo Levenshteinovo razdaljo = št. razlik med zaporedjema (vse kar ni ujemanje) v poravnavi.

s_1	C	–	A	–	C	A	T	A	G	A
s_2	T	G	A	T	T	A	–	A	G	–

ocena = −6

s_1	–	C	A	–	C	A	T	A	G	A
s_3	G	C	T	T	C	A	–	C	G	T

Ocena = −6

s_1	C	A	C	A	T	A	G	A
s_4	C	G	G	–	T	A	C	A

ocena = −4

s_2	T	G	–	A	T	T	A	–	A	G
s_3	–	G	C	T	T	C	A	C	G	T

ocena = −7

s_2	T	G	A	T	T	A	–	A	G
s_4	C	G	–	G	T	A	C	A	–

ocena = −5

s_3	G	C	T	T	C	A	C	G	T
s_4	–	C	G	G	T	A	C	–	A

ocena = −6

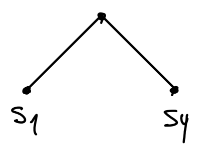
Iz ocen sestavimo matriko razdalj. Upoštevamo razdalja = |ocena|.

	s_1	s_2	s_3	s_4
s_1	0	6	6	4
s_2	6	0	7	5
s_3	6	7	0	6
s_4	4	5	6	0

Iz matrike začnemo graditi drevo. Uporabil bom metodo UPGMA.

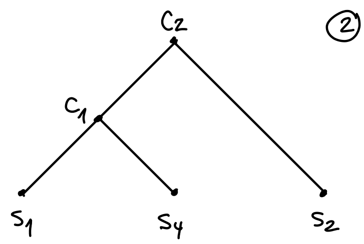
$C_1 = \{S_1, S_4\}$

1



$$d(C_1, S_2) = \frac{d(S_1,S_2) + d(S_4,S_2)}{|C_1|} = \frac{6+4}{2} = 5, \quad d(C_1, S_3) = \frac{d(S_1,S_3) + d(S_4,S_3)}{|C_1|} = \frac{6+6}{2} = 6$$

	C_1	S_2	S_3
C_1	0	5	6
S_2	5	0	7
S_3	6	7	0

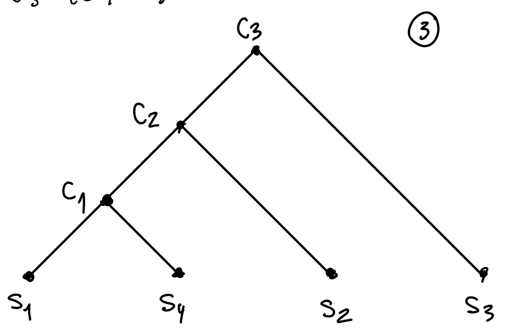


2

$C_2 = \{C_1, S_2\}$

Tu lahko že opazimo, da se bo S_3 vezal s C_2, zato računanje razdalj ni več potrebno.

$C_3 = \{C_2, S_3\}$



3

Vsaki točki določimo poravnavo in sicer najboljšo poravnavo zaporedje ene skupine z zaporedjem druge skupine (oz. poddrevesa).

1. Za C_1 imamo možni samo dve zaporedji, zato C_1 dodelimo poravnavo (S_1, S_4).

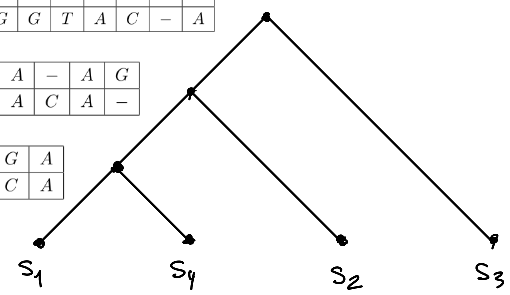
2. C_2 dodelimo poravnavo, ki zadostuje $\min\{C_1, S_2\} = \min\{d(S_1,S_2), d(S_4,S_2)\} = \min\{6, 5\} \Rightarrow (S_4, S_2)$.

3. C_3 dodelimo poravnavo $\min\{C_2, S_3\}$, kjer zaporedja za C_2 obravnavamo le S_4 in S_2.
$\min\{d(S_4, S_3), d(S_2, S_3)\} = \min\{6, 7\} \Rightarrow (S_4, S_3)$

s_3	G	C	T	T	C	A	C	G	T
s_4	$-$	C	G	G	T	A	C	$-$	A

s_2	T	G	A	T	T	A	$-$	A	G
s_4	C	G	$-$	G	T	A	C	A	$-$

s_1	C	A	C	A	T	A	G	A
s_4	C	G	G	$-$	T	A	C	A



Zaporedja ravnamo od vrha drevesa navzdol. Začnemo s poravnavo (s3, s4) in ji dodamo zaporedje s2, tako da dodamo vrzeli glede na to kako se s2 poravna z s2 in s3:

od (s4, s2)   od (s4, s2)

s3	G	C	T	-	T	C	A	C	G	T	-
s4	-	C	G	-	G	T	A	C	-	A	-
s2	-	T	G	A	T	T	A	-	-	A	G

od (s4, s3)   od (s4, s3)

Vsem dodamo še s1 in spet dodamo vrzeli glede na to kako se s4 poravna z s1 ter s2 in s3:

Končna poravnava:

s3	G	C	T	-	T	C	A	C	G	T	-
s4	-	C	G	-	G	T	A	C	-	A	-
s2	-	T	G	A	T	T	A	-	-	A	G
s1	-	C	A	C	A	T	A	G	-	A	-

od (s4, s3)

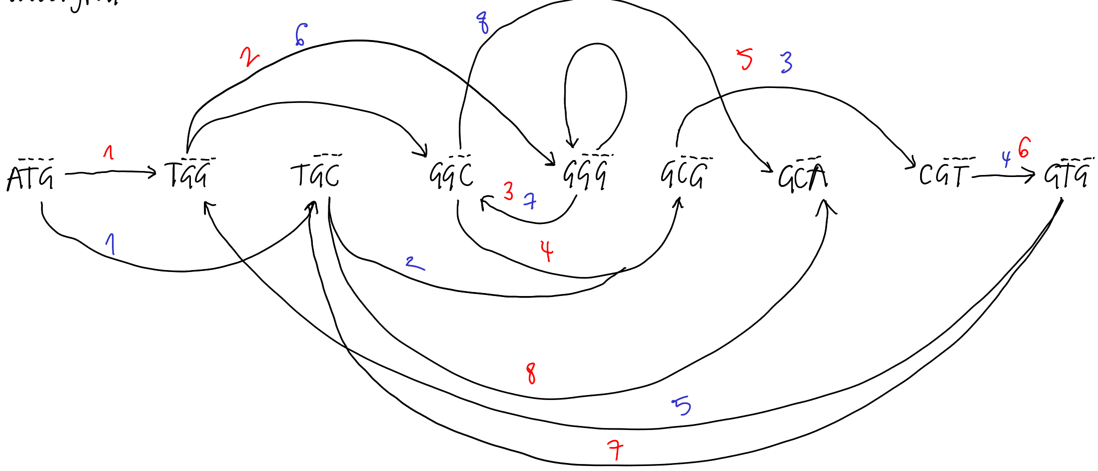
Naloga 2. – Rekonstrukcija zaporedij

Dani so naslednji podatki za problem rekonstrukcije zaporedij: $\ell = 3$, multimnožica zaporedij: $S = \{\mathtt{ATG}, \mathtt{CGT}, \mathtt{GCA}, \mathtt{GCG}, \mathtt{GGC}, \mathtt{TGG}, \mathtt{GTG}, \mathtt{TGC}, \mathtt{GGG}\}$.

a) Uporaba hamiltonskih poti

Hamiltonska pot: pot, ki obišče vsako točko natanko enkrat.

Multigraf:



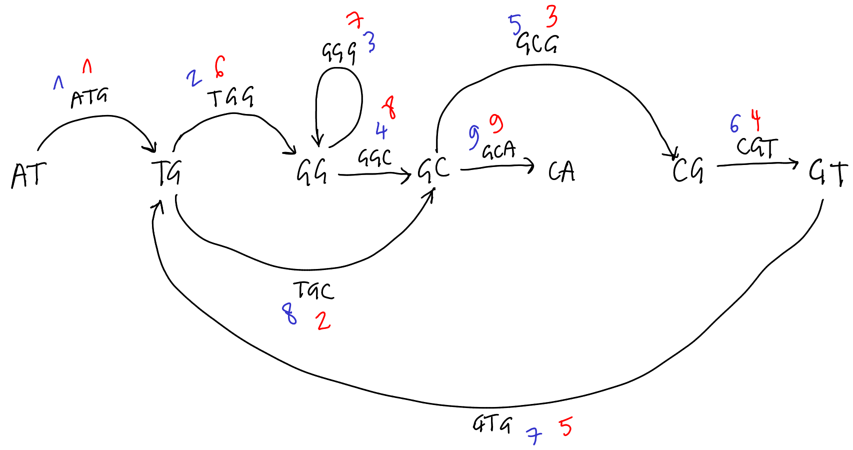
Možni poti:

A T G C G T G G G C A

A T G G G C G T G C A

b) Uporaba Eulerjevih sledi $S = \{\mathtt{ATG}, \mathtt{CGT}, \mathtt{GCA}, \mathtt{GCG}, \mathtt{GGC}, \mathtt{TGG}, \mathtt{GTG}, \mathtt{TGC}, \mathtt{GGG}\}$.

Eulerjeva sled: sled, ki gre po vsaki povezavi natanko enkrat.



Možni poti:

A T G G G C G T G C A

A T G C G T G G G C A

Naloga 3. – Aditivna filogenija

Obravnavajte naslednje tri matrike

M_1	A	B	C	D
A	0	2	6	8
B	2	0	6	9
C	6	6	0	8
D	8	10	8	0

M_2	A	B	C	D
A	0	2	6	8
B	2	0	7	9
C	6	7	0	8
D	8	9	8	0

M_3	A	B	C	D
A	0	5	6	8
B	5	0	10	6
C	6	10	0	7
D	8	6	7	0

(a) Za vsako od njih določite, ali je aditivna ali ne z uporabo pogoja štirih točk. *(3 točke)*

(b) Uporabite algoritem ADITIVNA FILOGENIJA na matriki M_2. Zapišite vse korake algoritma. *(5 točk)*

a)

M_1	A	B	C	D
A	0	2	6	8
B	2	0	6	9
C	6	6	0	8
D	8	10	8	0

$D_{AB} + D_{CD} = 2 + 8 = 10$

$D_{AC} + D_{BD} = 6 + 10 = 16$

$D_{AD} + D_{BC} = 8 + 6 = 14$

$\Rightarrow$ Ni aditivna.

M_2	A	B	C	D
A	0	2	6	8
B	2	0	7	9
C	6	7	0	8
D	8	9	8	0

$D_{AB} + D_{CD} = 2 + 0 = 2$

$D_{AC} + D_{BD} = 6 + 9 = 15$

$D_{AD} + D_{BC} = 8 + 7 = 15$

$\Rightarrow$ Je aditivna.

M_3	A	B	C	D
A	0	5	6	8
B	5	0	10	6
C	6	10	0	7
D	8	6	7	0

$D_{AB} + D_{CD} = 5 + 7 = 12$

$D_{AC} + D_{BD} = 6 + 6 = 12$

$D_{AD} + D_{BC} = 8 + 10 = 18$

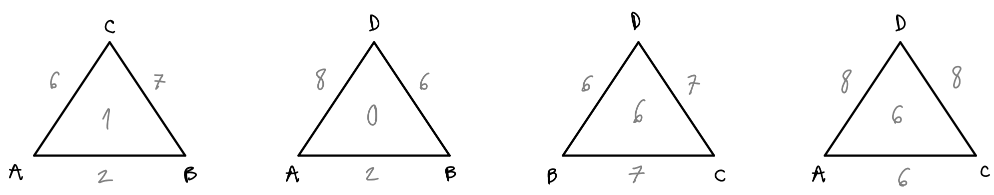
$\Rightarrow$ Ni aditivna.

b) Potek algoritma:

1. Izberemo trojico z najmanjšo skupno razdaljo
2. Določimo $\delta = 1/2 \cdot$ razdalja iz (1)
3. Vsem (razen diagonalnim) elementom v matriki odštejemo 2δ
4. Znebimo se ene instance iz matrike

M_2	A	B	C	D
A	0	2	6	8
B	2	0	7	9
C	6	7	0	8
D	8	9	8	0

① 1.



$\rightarrow$ Trojica ABD. Veljati mora:

$D_{AD} = D_{AB} + D_{BD}$

$8 = 2 + 6$ ✓

skupna razdalja = Σ krajši dve stranici − tretja

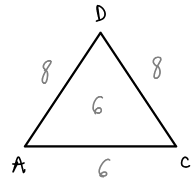
2. $\delta = 1/2 \cdot 0 = 0$

3. Matrika ostane enaka ($2\delta = 0$)

4. Znebimo se vmesne instance = B :

	A	C	D
A	0	6	8
C	6	0	8
D	8	8	0

② 1.



$\Rightarrow D_{AD} = D_{AC} + D_{CD}$

$8 = 6 + 8$!

Ne drži.

...

Hm

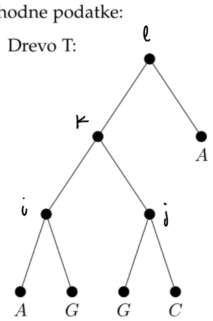
kaj pa zdaj

4. Rešite mali problem varčnosti za naslednje vhodne podatke:

Cene mutacij:

δ	A	C	G	T
A	0	1	2	3
C	1	0	4	3
G	2	4	0	5
T	3	3	5	0

Drevo T:



Iščemo poravnave zaporedij notranja vozlišča.

Najprej vsakemu listu določimo tabelo:

A:

A	C	G	T
0	∞	∞	∞

C:

A	C	G	T
∞	0	∞	∞

G:

A	C	G	T
∞	∞	0	∞

T:

A	C	G	T
∞	∞	∞	0

Točki i določimo tabelo glede na A in G tabeli:

vzamemo min vrednost iz vsakega izračuna

vrstica iz matrike

	A	C	G	T		A	C	G	T	
A:	0	∞	∞	∞	G:	∞	∞	0	∞	
A:	0	1	2	3 +	A:	0	1	2	3 +	
	0	∞	∞	∞		∞	∞	2	∞	= 0 + 2 = 2

	A	C	G	T		A	C	G	T	
A:	0	∞	∞	∞	G:	∞	∞	0	∞	
C:	1	0	4	3 +	C:	1	0	4	3 +	
	1	∞	∞	∞		∞	∞	0	∞	= 1 + 0 = 1

	A	C	G	T		A	C	G	T	
A:	0	∞	∞	∞	G:	∞	∞	0	∞	
G:	2	4	0	5 +	G:	2	4	0	5 +	
	2	∞	∞	∞		∞	∞	0	∞	= 2 + 0 = 2

	A	C	G	T		A	C	G	T	
A:	0	∞	∞	∞	G:	∞	∞	0	∞	
T:	3	3	5	0 +	T:	3	3	5	0 +	
	3	∞	∞	∞		∞	∞	0	∞	= 3 + 0 = 3

$\Rightarrow$ i:

A	C	G	T
2	1	2	3

Točki j določimo glede na G in C:

	A	C	G	T		A	C	G	T	
G:	∞	∞	0	∞	C:	∞	0	∞	∞	
A:	0	1	2	3 +	A:	0	1	2	3 +	
	∞	∞	2	∞		∞	1	∞	∞	= 2 + 1 = 3

	A	C	G	T		A	C	G	T	
G:	∞	∞	0	∞	C:	∞	0	∞	∞	
C:	1	0	4	3 +	C:	1	0	4	3 +	
	∞	∞	0	∞		∞	0	∞	∞	= 0 + 0 = 0

	A	C	G	T		A	C	G	T	
G:	∞	∞	0	∞	C:	∞	0	∞	∞	
G:	2	4	0	5 +	G:	2	4	0	5 +	
	∞	∞	0	∞		∞	4	∞	∞	= 0 + 4 = 4

	A	C	G	T		A	C	G	T	
G:	∞	∞	0	∞	C:	∞	0	∞	∞	
T:	3	3	5	0 +	T:	3	3	5	0 +	
	∞	∞	0	∞		∞	3	∞	∞	= 0 + 3 = 3

j:

A	C	G	T
3	0	4	3

$\Uparrow$

Točki k določimo glede na i in j tabeli:

	A	C	G	T		A	C	G	T	
i:	2	1	2	3	j:	3	0	3	4	
A:	0	1	2	3 +	A:	0	1	2	3 +	
	2	2	4	6		3	1	5	7	= 2 + 1 = 3

	A	C	G	T		A	C	G	T	
i:	2	1	2	3	j:	3	0	3	4	
C:	1	0	4	3 +	C:	1	0	4	3 +	
	3	1	6	6		4	0	7	7	= 1 + 0 = 1

	A	C	G	T		A	C	G	T	
i:	2	1	2	3	j:	3	0	3	4	
G:	2	4	0	5 +	G:	2	4	0	5 +	
	4	5	2	8		5	4	3	9	= 2 + 3 = 5

	A	C	G	T		A	C	G	T	
i:	2	1	2	3	j:	3	0	3	4	
T:	3	3	5	0 +	T:	3	3	5	0 +	
	5	4	7	3		6	3	8	4	= 3 + 3 = 6

$\Rightarrow$ k:

A	C	G	T
3	1	5	6

Točki l določimo glede na A in k tabeli:

	A	C	G	T
A:	0	∞	∞	∞
A:	0	1	2	3 +
	0	∞	∞	∞

	A	C	G	T
k:	3	1	5	6
A:	0	1	2	3 +
	3	2	7	9

$= 0 + 2 = 2$

	A	C	G	T
A:	0	∞	∞	∞
C:	1	0	4	3 +
	1	∞	∞	∞

	A	C	G	T
k:	3	1	5	6
C:	1	0	4	3 +
	3	1	9	9

$= 1 + 1 = 2$

	A	C	G	T
A:	0	∞	∞	∞
G:	2	4	0	5 +
	2	∞	∞	∞

	A	C	G	T
k:	3	1	5	6
G:	2	4	0	5 +
	5	5	5	11

$= 2 + 5 = 7$

	A	C	G	T
A:	0	∞	∞	∞
T:	3	3	5	0 +
	3	∞	∞	∞

	A	C	G	T
k:	3	1	5	6
T:	3	3	5	0 +
	6	4	10	6

$= 3 + 4 = 7$

→ l:

A	C	G	T
2	2	7	7

Vsaki točki moramo zdaj določiti znak. Načeloma izberemo najmanjšo vrednost v tabeli