Univerza v Ljubljani, Fakulteta za elektrotehniko
Podiplomski magistrski študijski program 2. stopnje Elektrotehnika
Biomedicinska tehnika - Biomedicinska informatika

NAVODILA

Vaja 8: Ujemanje zaporedij

Pripravila: Žiga Bizjak & Tomaž Vrtovec

Navodila

Needleman-Wunsch-ev algoritem je uveljavljen pristop za določanje globalnega optimalnega ujemanja zaporedij (npr. nukleotidov v DNK ali aminokislin v beljakovinah). Algoritem je poseben primer metode *dinamičnega programiranja*, s pomočjo katere lahko rešimo relativno kompleksne probleme tako, da jih razdelimo na lažje podprobleme.

1. Napišite funkcijo za določanje matrike ocen M in poti T za ujemanje zaporedij a in b:

```
def computeMatrices(iSeqA, iSeqB, iSubS, iSubM, iGapP):
    # ...
    # your code goes here
    # ...
    return oScrM, oTrcM
```

   kjer vhodna argumenta `iSeqA` in `iSeqB` predstavljata zaporedji a in b, `iSubS` predstavlja zastopane znake v izbranem vrstnem redu (npr. `'AGCT'`), `iSubM` predstavlja substitucijsko matriko S z znaki v enakem vrstnem redu, `iGapP` pa predstavlja kazen za vrinjeno mesto P. Izhodna argumenta `oScrM` in `oTrcM` predstavljata matriko ocen M in matriko poti T.

   Matriko ocen M ustrezno inicializirajte z začetnimi vrednostmi (ničle), matriko poti T pa z začetnimi smermi. Oceno $M(i,j)$ in smer $T(i,j)$ določite s pomočjo rekurzivne formule:

$$M(i,j) = max \begin{cases} M(i-1,j-1) + S\big(a(i),b(j)\big) & \nwarrow \text{ (smer diagonala oz. D)} \\ M(i,j-1) + P & \leftarrow \text{ (smer levo oz. L)} \\ M(i-1,j) + P & \uparrow \text{ (smer gor oz. U)} \end{cases}$$

   V primeru, da je več vrednosti hkrati največjih, v matriko poti T vpišete samo eno smer, pri čemer se držite predpisanega vrstnega reda, in sicer najprej smer diagonala ($\nwarrow$ oz. D), nato smer levo ($\leftarrow$ oz. L) in nazadnje smer gor ($\uparrow$ oz. U).

2. Napišite funkcijo za določanje optimalno ujemajočih se zaporedij:

```
def computeSequences(iSeqA, iSeqB, iTrcM):
    # ...
    # your code goes here
    # ...
    return oSeqA, oSeqB
```

   kjer vhodna argumenta `iSeqA` in `iSeqB` predstavljata zaporedji a in b, `iTrcM` pa matriko poti T. Izhodna argumenta `oSeqA` in `oSeqB` predstavljata optimalno ujemajoči se zaporedji a in b, ki ju poiščete s pomočjo sledenja glede na smeri v matriki poti T, pri čemer s sledenjem pričnete v zadnjem elementu matrike.

Preverite implementacijo algoritma tako, da določite optimalno ujemanje DNK zaporedij a = `GGATCGA` in b = `GAATTCAGTTA`, za katere je bila rešitev podana pri pripravah na

laboratorijske vaje. Uporabite substitucijsko matriko S in kazen za vrinjeno mesto P:

$S =$

	A	G	C	T
A	2	−1	−1	−1
G	−1	2	−1	−1
C	−1	−1	2	−1
T	−1	−1	−1	2

$P = -2$

$S^* =$

	A	G	C	T
A	2	1	−1	−1
G	1	2	−1	−1
C	−1	−1	2	1
T	−1	−1	1	2

$P^* = 0$

Vprašanja

Odgovore na sledeča vprašanja zapišite v poročilo, v katerega vstavite zahtevane izrise in programske kode.

1. Napišite funkcijo za izračun ocene optimalnega ujemanja:

```
def computeScore(iSeqA, iSeqB, iSubS, iSubM, iGapP):
    # ...
    # your code goes here
    # ...
    return oScr
```

   kjer vhodna argumenta `iSeqA` in `iSeqB` predstavljata optimalno ujemajoči se zaporedji a in b, `iSubS` predstavlja znake v izbranem vrstnem redu (npr. `'AGCT'`), `iSubM` predstavlja substitucijsko matriko S z znaki v enakem vrstnem redu, `iGapP` pa predstavlja kazen za vrinjeno mesto P. Izhodni argument `oScr` predstavlja oceno optimalnega ujemanja.

   Priložite programsko kodo funkcije `computeScore()`.

2. Določite optimalno ujemanje DNK zaporedij a = ACA in b = CGACT, pri čemer uporabite substitucijsko matriko S in kazen za vrinjeno mesto P. Zapišite tudi matriko ocen M, matriko poti T z označeno optimalno potjo ter oceno optimalnega ujemanja.

3. Določite optimalno ujemanje DNK zaporedij a = CTCTAGCATTAG in b = GTGCACCCA, pri čemer uporabite substitucijsko matriko S in kazen za vrinjeno mesto P. Zapišite tudi oceno optimalnega ujemanja.

4. Določite optimalno ujemanje zaporedij iz vprašanja št. 2, pri čemer uporabite substitucijsko matriko S^* in kazen za vrinjeno mesto P^*. Zapišite tudi matriko ocen M, matriko poti T z označeno optimalno potjo ter oceno optimalnega ujemanja.

5. Določite optimalno ujemanje zaporedij iz vprašanja št. 3, pri čemer uporabite substitucijsko matriko S^* in kazen za vrinjeno mesto P^*. Zapišite tudi oceno optimalnega ujemanja.

Dodatek

Odgovore na sledeče probleme ni potrebno prilagati k poročilu, prispevajo pa naj k boljšemu razumevanju vsebine.

Pri obstoječi implementaciji algoritma smo zanemarili dejstvo, da lahko obstaja več optimalnih ujemanj danih zaporedij, ki bi izhajala iz več možnih izbir v matriki poti T zaradi hkratnih največjih vrednosti v rekurzivni formuli za iskanje vrednosti matrike ocen M. Popravite obstoječo implementacijo tako, da bo omogočala določanje vseh obstoječih optimalnih ujemanj zaporedij.