Karakterláncok és pontos illesztés

Lehotay-Kéry Péter
lepuaai at inf dot elte dot hu

Ben Langmead diasora alapján
(www.langmead-lab.org/teaching-materials)

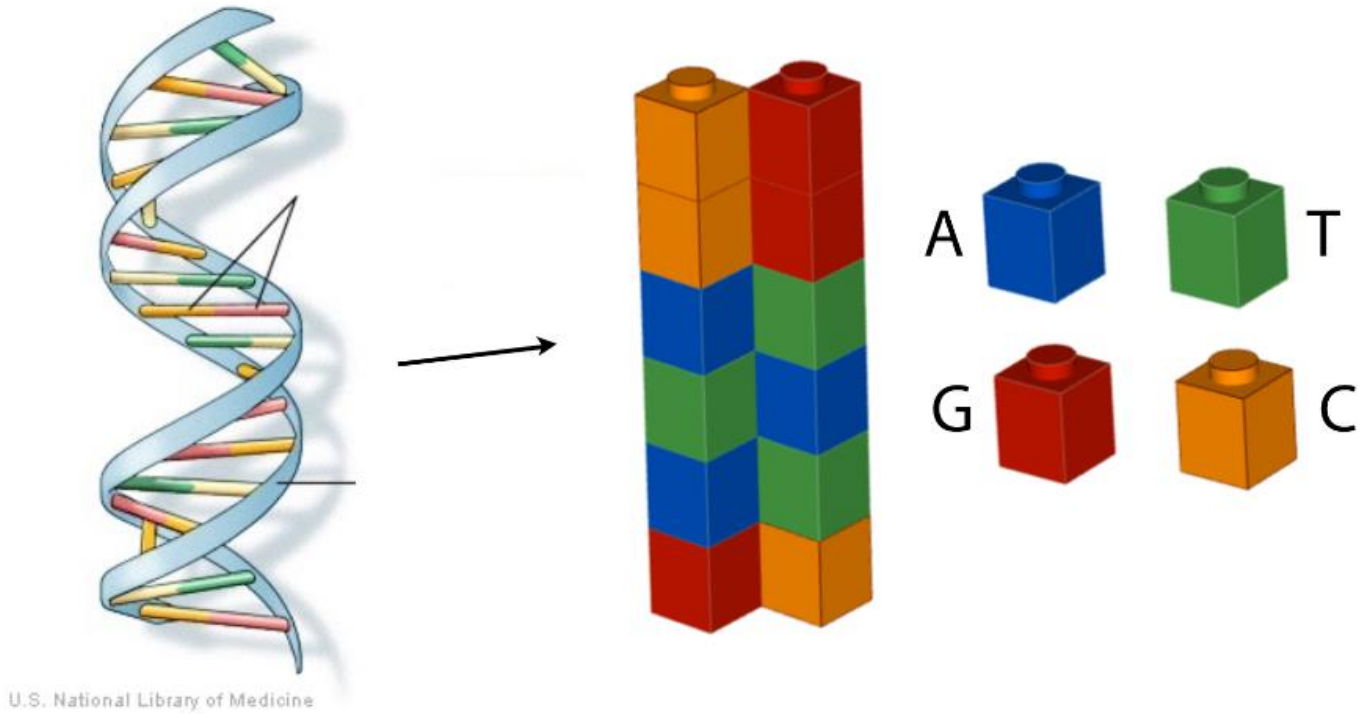
Karakterlánc (String) definiciók

- S string egy véges, rendezett karakterlista
- A karakterek egy Σ ábécéből kerülnek ki, $|\Sigma|$ eleme van.
  - Nukleinsav ábécé: { A, C, G, T }
  - Aminosav ábécé: { A, R, N, D, C, E, Q, G, H, I, L, K, M, F, P, S, T, W, Y, V }
- S hossza $|S|$, ahány karaktere van S-nek.
- ϵ az üres karakterlánc. $|\epsilon|=0$

Karakterlánc (String) definiciók

- S és T Σ ábécé fölötti karakterláncok, konkatenációjuk S karakterei, melyeket T karakterei követnek: ST.
- S részstringje T-nek, ha léteznek u és v stringek, melyekre T=uSv
- S T prefixe, ha létezik u string, melyre T=Su.
- S T szuffixe, ha létezik u string, melyre T=uS.
- S valódi prefix (szuffix), ha se S, se u nem üres.
- Részszekvencia hasonló részstringhez, de a karakterek nem feltétlenül követik egymást: “cant” részszekvenciája “con**ca**te**n**a**t**e”-nek, de nem részstringje.

Forditott komplemens

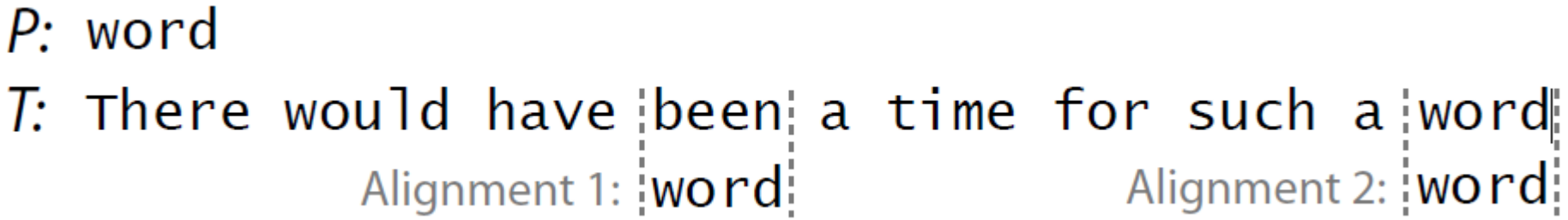


Double stranded DNA (double helix)

Double stranded DNA (lego version)

Pontos illesztés

- Olyan helyeket keresünk, ahol P minta előfordul részstringként T szövegben. Minden ilyen hely egy előfordulás, vagy illeszkedés.
- Legyen n=|P| és m=|T| és n≤m.
- Illesztés egy módja annak, hogy P karaktereit T karaktereivel szembe helyezzük. Ez lehet előfordulás vagy nem.



Pontos illesztés

P: word

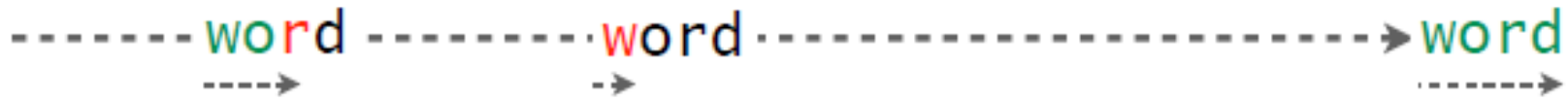
T: There would have been a time for such a word

word word word word word word word word word
word word word word word word word word
word word word word word word word word
word word word word word word word word
word word word word word word word word

One occurrence

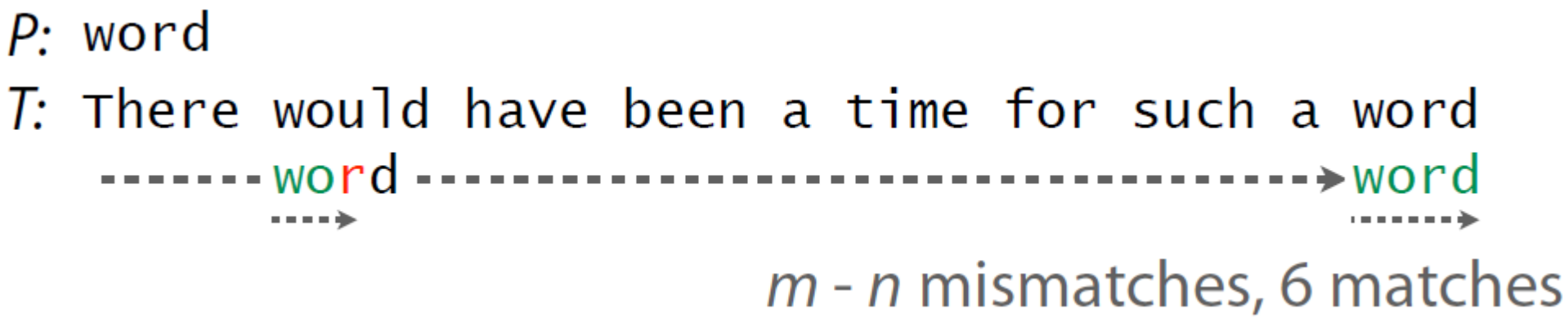
P: word

T: There would have been a time for such a word



Pontos illesztés: naive algoritmus

- Mennyi illesztés lehetséges adott n-re és m-re?
- Mennyi karakter összehasonlitás lehetséges maximum?
- Minimum?
- Mennyi karakterösszehasonlitás történik az alábbi példában?

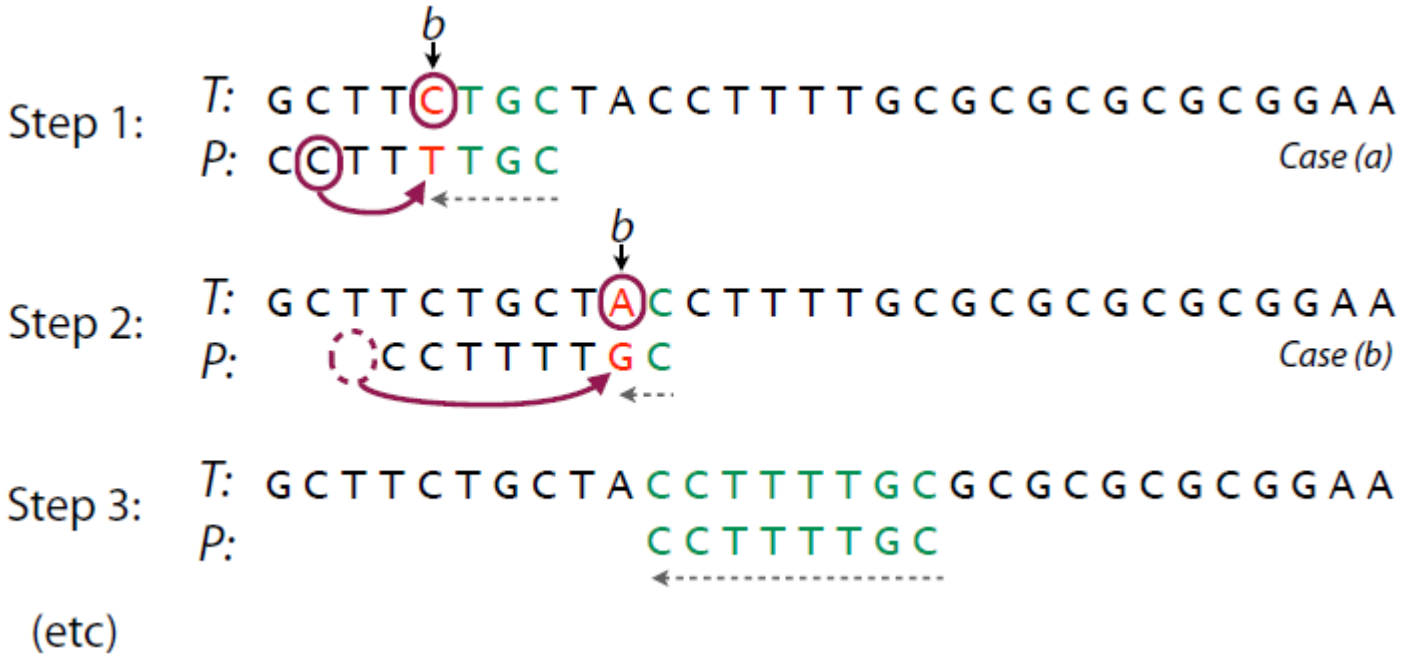


Boyer-Moore

- Hagyjuk ki a következő illesztéseket, melyek biztosan nem fognak egyezni:
- Rossz karakter szabály: Ha nem egyezik a karakter, használjuk ki, hogy illesztéseket hagyjunk ki.
- Jó szuffix szabály: Ha egyezik néhány karakter, használjuk ki, hogy illesztéseket hagyjunk ki.
- Hosszabb kihagyásokért: Próbáljunk illesztéseket egy irányba, karakter összehasonlitást a másikba.

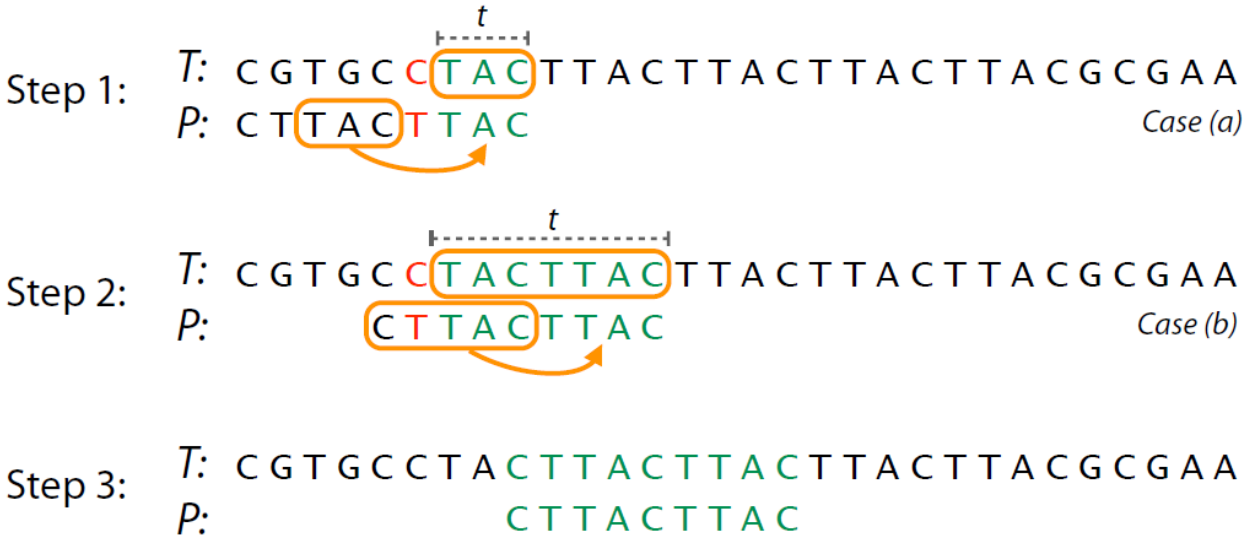
Boyer-Moore: Rossz karakter szabály

- Ha nem egyezik egy karakter, legyen b a nem egyező karakter T-ben. Hagyjuk ki az illesztéseket, amig
  - a) b nem egyezik a vele szembe levővel P-ben
  - B) P át nem halad b-n.



Boyer-Moore: Jó szuffix szabály

- Legyen t T részstringje, mely egyezett P szuffixével. Hagyjuk ki az illesztéseket, amig
  - a) t nem egyezik a szemben levő P karakterekre
  - b) P prefixe nem egyezik t szuffixével

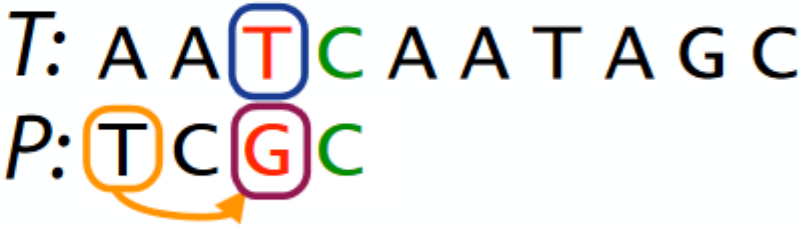


Boyer-Moore: előfeldolgozás

- Előre kiszámitott lépések a rossz karakter szabályra, P=TCGC

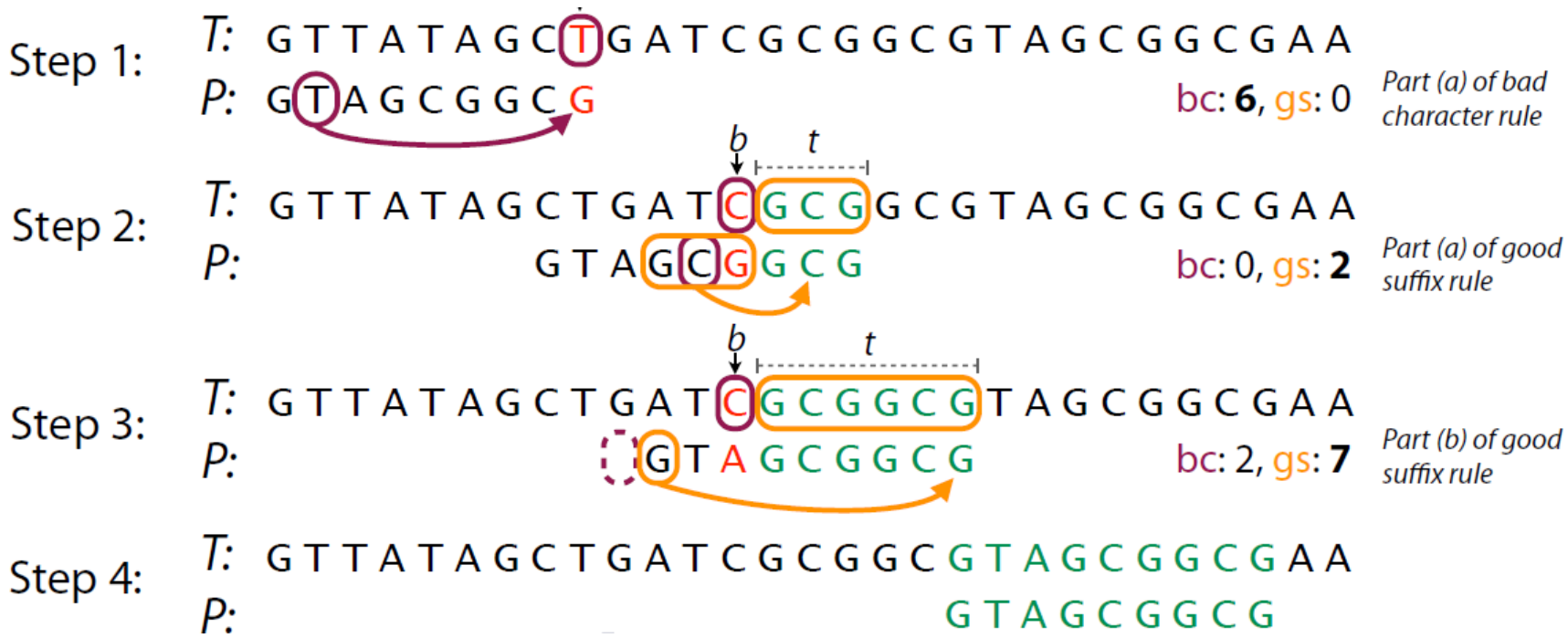
P

Σ	T	C	G	C
A	0	1	0	3
C	0	-	2	-
G	0	1	-	0
T	-	0	1	2



Boyer-Moore: Tegyük össze

- Minden illesztésre használjuk a rossz karakter, vagy jó szuffix szabályt, amelyik több illesztést hagy ki.



Feladat

- Választható feladatok:
  1. Irjuk át a naive és Boyer-Moore illesztést úgy, hogy számolja össze, hány illesztést és hány karakter összehasonlitást végez! (2 szám lesz az eredmény az egyik lefutásakor és 2 szám a másik lefutásakor)
  2. Irjuk át a naive illesztést úgy, hogy a minta forditott komplemenseit is megtalálja! (Az eredmény az illeszkedő poziciók és a forditott komplemens illeszkedő poziciói egy tömbben, szigorúan monoton növekvő sorrendben.)
- Készitsetek egy notebookot, mely a házi feladatokat fogja tartalmazni, ezt osszátok meg az e-mail cimemmel: lepuaai at inf dot elte dot hu