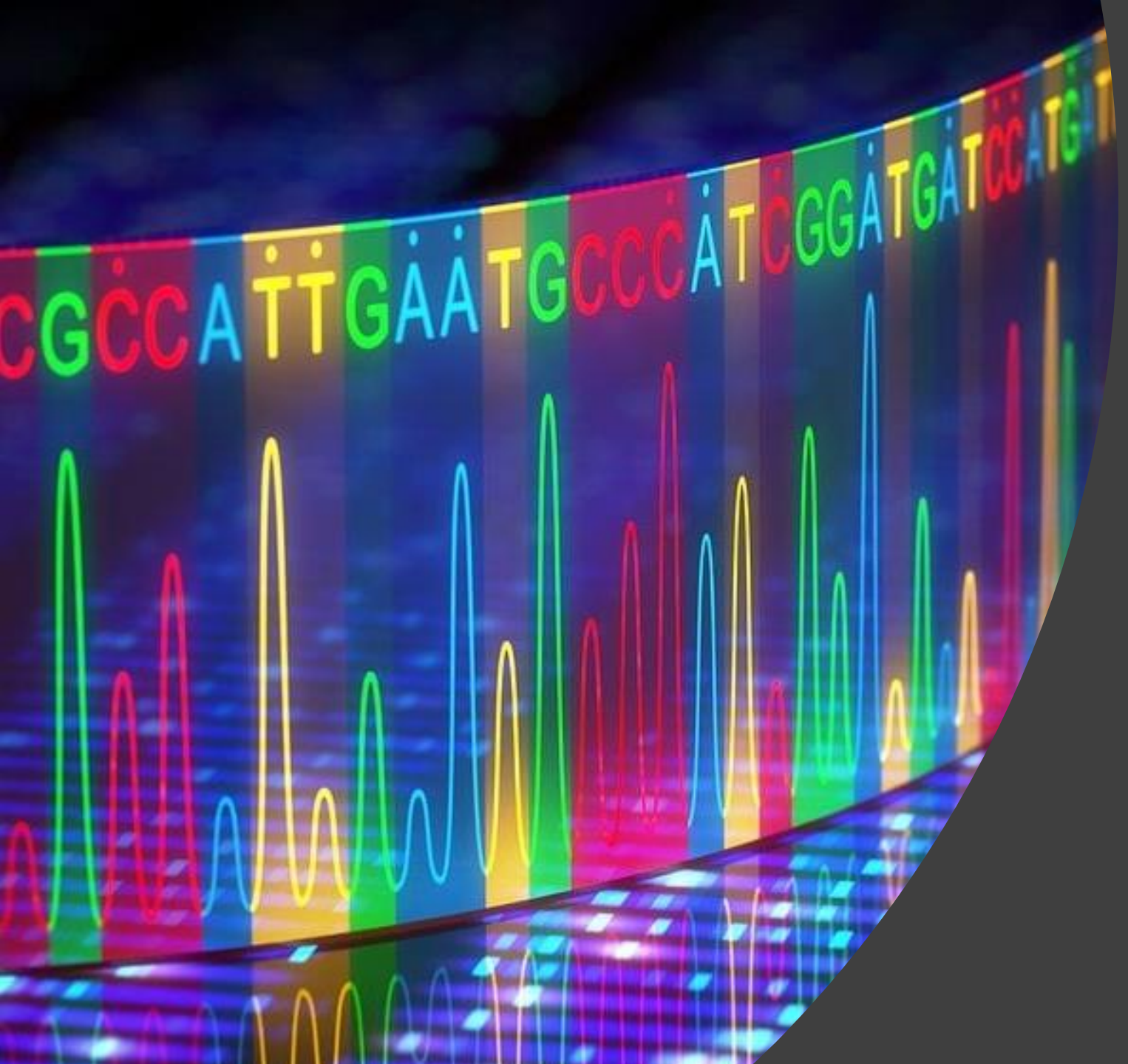


Hirschbergův algoritmus

Programování v bioinformatice

MPC – PRG 2021/2022

Vyučující:
Ing. Kateřina Jurečková(garant)
Ing. et Ing. Jana Schwarzerová, MSc

Opakování

- Co je to algoritmus?

(určovat jeho výpočetní náročnost, podle čeho se hodnotí apod.)

- Co je to rekurze?

(rozdíl mezi rekurzí a iterací)

⇒Hirschbergův algoritmus

http://users.monash.edu/~lloyd/tildeAlgDS/Dynamic/Hirsch/

Hirschbergův algoritmus

- **Daniel S. Hirschberg**

- Dynamický programovací algoritmus (DPA)
  - obecně použitelný pro optimální zarovnání sekvencí

- Účel: nalézt optimální zarovnání sekvence mezi dvěma řetězci

Hirschbergův algoritmus

- Hirschbergův algoritmus je jednoduše popsán jako prostorově efektivnější verze algoritmu Needleman – Wunsch

- Praktické využití – výpočetní biologie
  – slouží k nalezení maximálního globálního zarovnání sekvencí DNA a proteinů

Měli byste vědět co je N-W algoritmus a rozdíl mezi globálním a lokálním zarovnáním [viz ABIN – Bioinformatika, bakalářský program]

Vsuvka – Needleman-Wunsch Algoritmus

- **Saul B. Needleman & Christian D. Wunsch**
- Algoritmus dynamického programování s cílem provést globální zarovnání sekvencí

H(i, j) = hodnota pole v matici na souřadnicích ***i, j***

d = sankce za vložení mezery (záporné číslo, nebo 0)

F(i, j) = hodnota skóre za shodu/neshodu znaků v matici na souřadnicích *i, j*

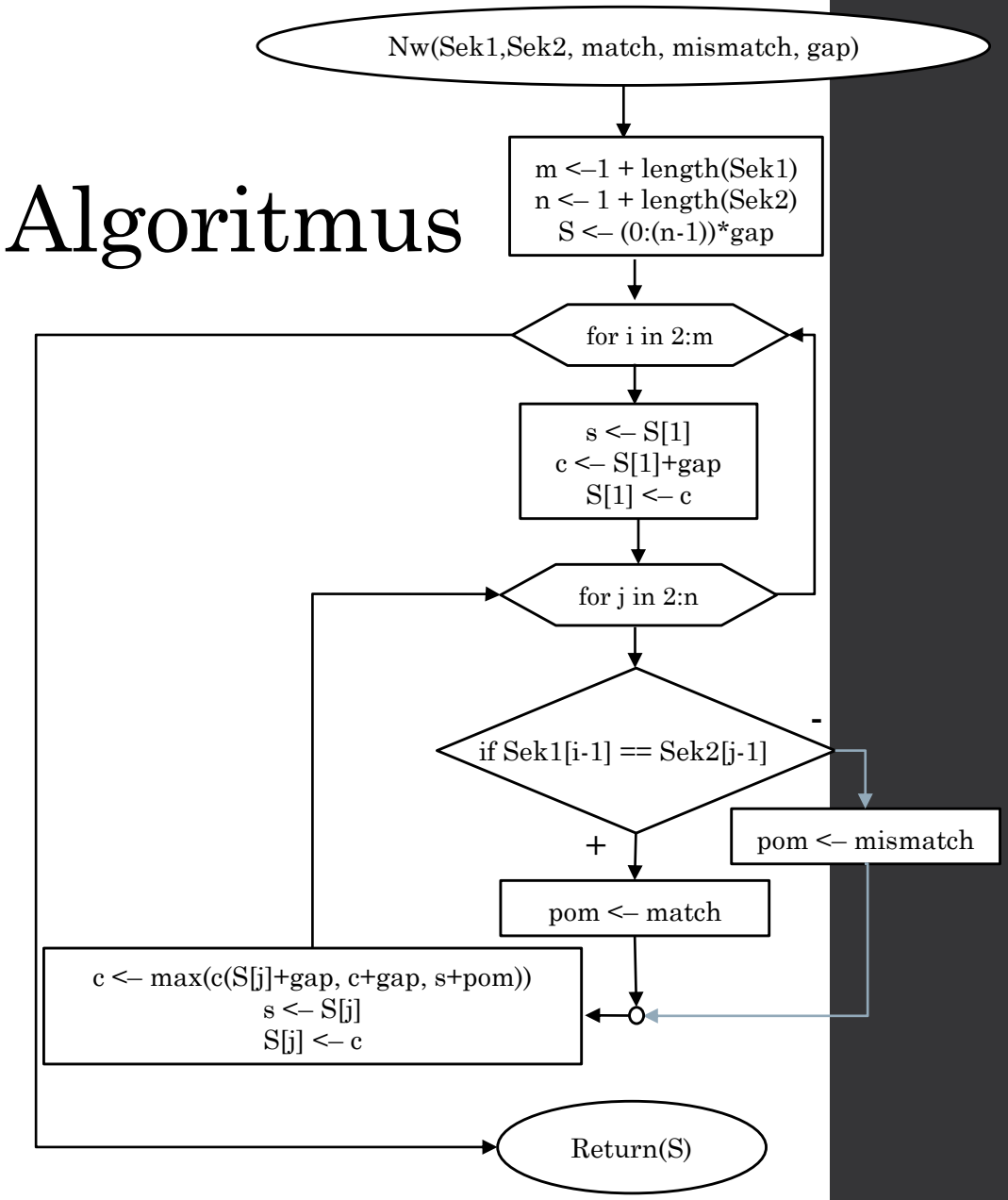
Vsuvka – Needleman-Wunsch Algoritmus

1) Inicializace nuly do horního rohu matice
2) Do každého dalšího pole:

$$\max \begin{cases} H(i-1, j) + d \\ H(i, j-1) + d \\ H(i-1, j-1) + F(i, j) \end{cases}$$

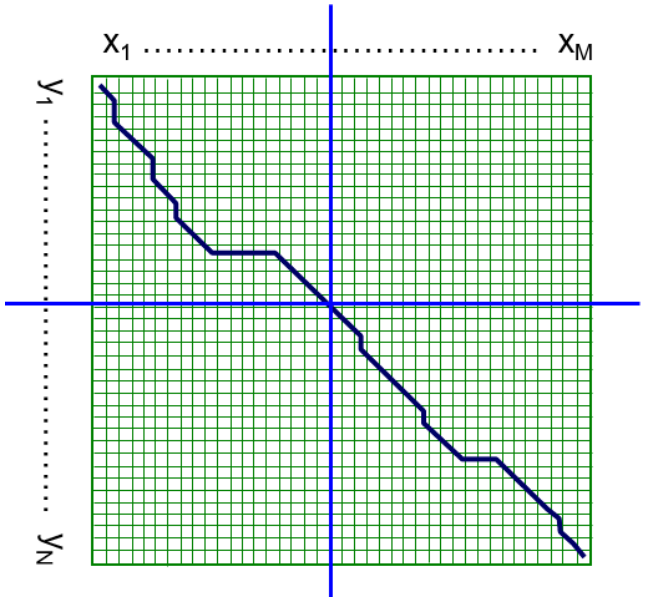
3) Po zapsání všech hodnot do matice nalezneme zpětnou cestu

-	-	A	T	C	G	A	C
-	0	-4	-8	-12	-16	-20	-24
C	-4	-3	-7	-3	-7	-11	-15
A	-8	1	-3	-7	-6	-2	-6
T	-12	-3	6	2	-2	-6	-5
A	-16	-7	2	3	-1	3	-1
C	-20	-11	-2	-1	0	-1	8



Vsuvka

- ABIN – N-W algoritmus ✓
- MPC-PRG – Cvičení 2 – náročnost algoritmu ✓



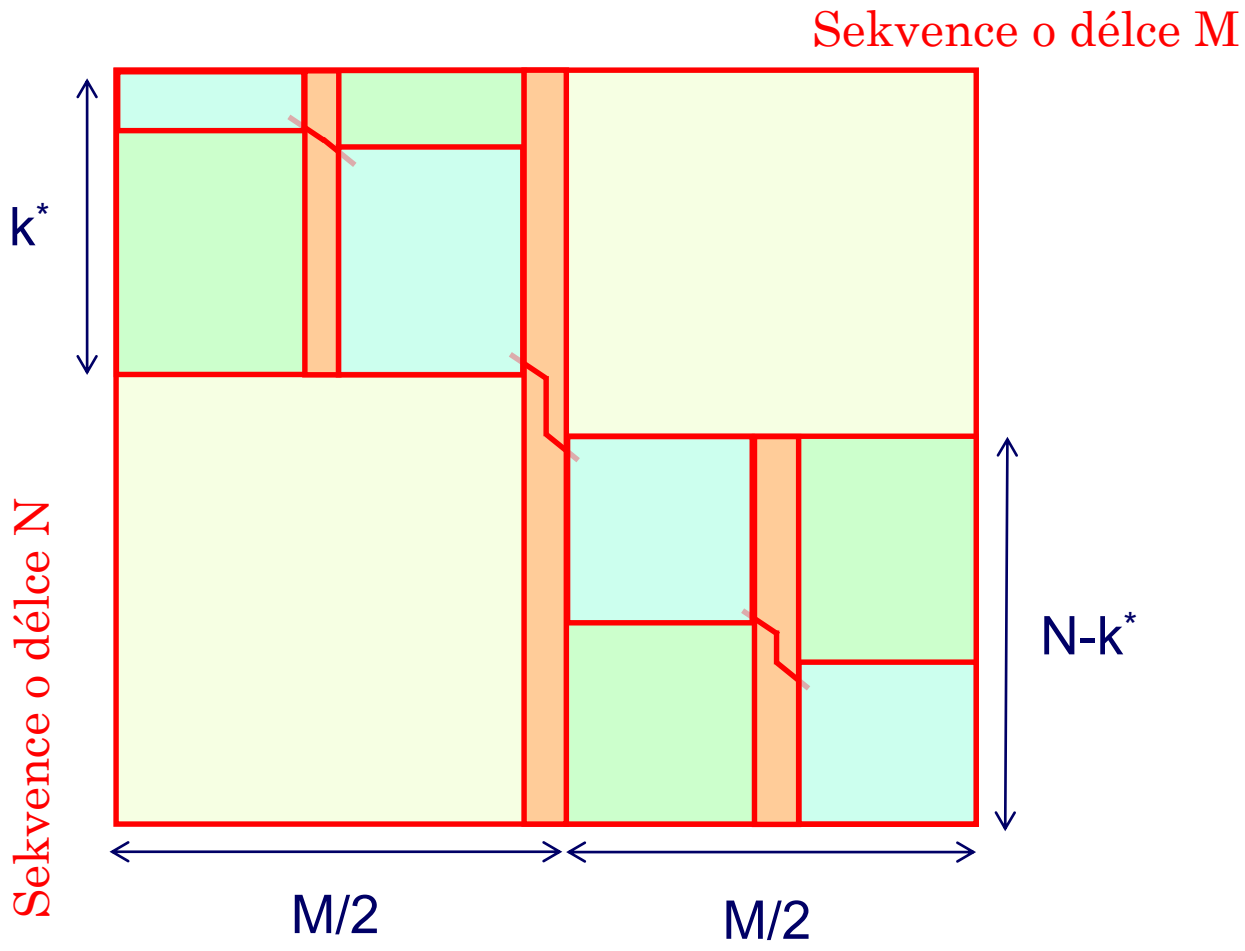
Časová náročnost algoritmu – O(M*N)
Prostorová náročnost algoritmu – O(M*N)

=> N-W není moc efektivní
… Co tak použít nějakou rekurzi?

- MPC-PRG – Cvičení 4 – Rekurze ✓

N-W Algoritmus + princip rekurzí
↓
efektivnějšího algoritmu

Hirschbergův algoritmus



Příklad (z přednášky)

Zadání:

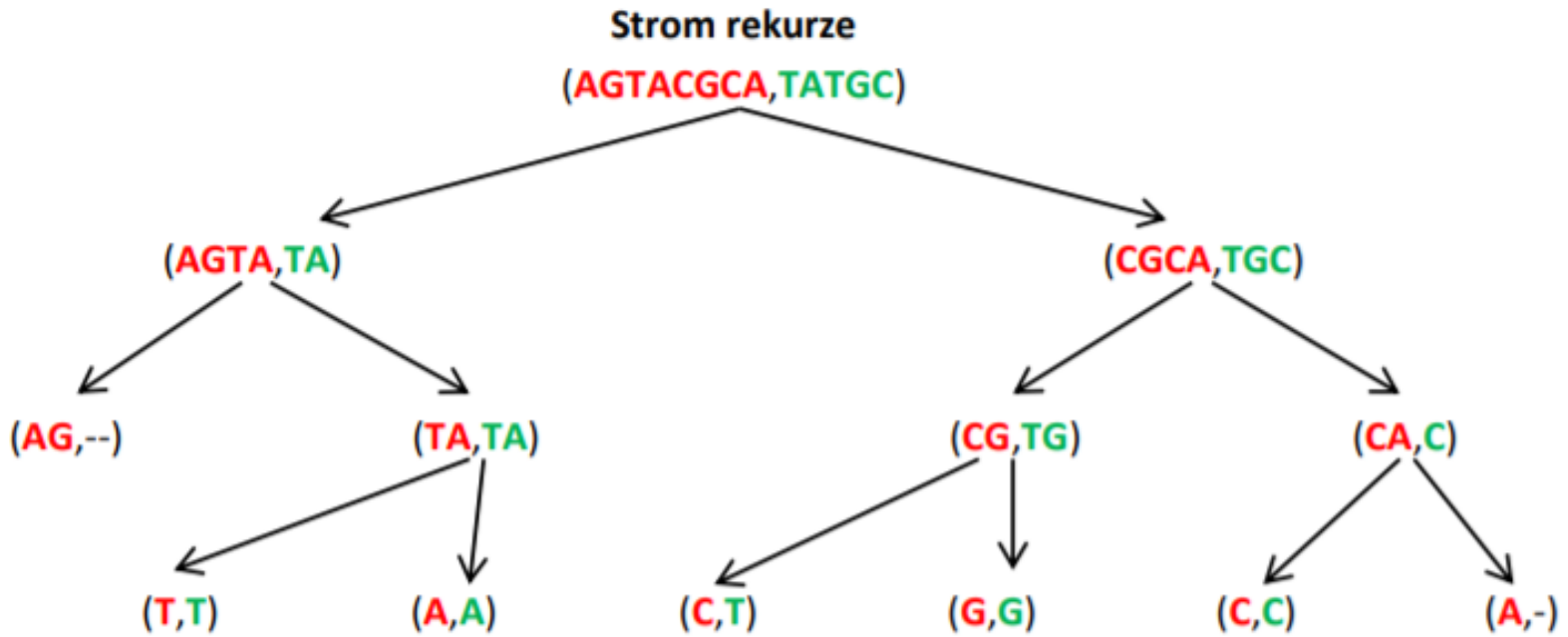
X = AGTACGCA

Y = TATGC

match = 2; mismatch = -1; gap = -2

Příklad

Hirschberg(AGTACGCA, TATGC)



Hirschbergův algoritmus – souhrn

- Máme sadu řetězců označených jako ***x, y*** a jejich sub-sekvenci ***u***.

Def. problém: hledáme nejdelší společnou sub-sekvenci $u = u_1 \ldots u_k$,
danou řetězci $x = x_1\, x_2 \ldots x_M$, $y = y_1\, y_2 \ldots y_N$,

- Algoritmus:

$$F(i, j) = \max \begin{cases} F(i-1, j) \\ F(i, j-1) \\ F(i-1, j-1) + [1 - \text{když } x_i = y_j;\ 0 - \text{cokoli jiného}] \end{cases}$$

Ptr(i, j) = stejné jako N-W algoritmus

Ukončení: trasa zpět z Ptr(M, N), a přiřazení ***u*** kdykoli, když
Ptr(i, j) = DIAG and $F(i-1, j-1) < F(i, j)$

Souhrn úkolů pro cvičení 4

1) Pochopit Hirschbergův algoritmus
2) Projít si ZNOVU příklady
   – z přednášky –pdf viz týden 3
   – ze cvičení – viz týden 4 (doptat se k věcem, které mi nejsou jasné)
3) Zkontrolovat zda jsem NEPODCENIL/A bod 1 + 2 !!!
4) Implementovat Hirschbergův algoritmus do libovolného programovacího jazyk (**R** - doporučeno / Matlab / Python etc.)

[Nápověda – https://en.wikipedia.org/wiki/Hirschberg%27s_algorithm]

V časovém rozmezí cvik byste ovšem měli být MINIMÁLNĚ schopni vyřešit aspoň sub-problémy:
dělit sekvence + najít další bod dělení

Každým rokem je největší problém se správným předáváním proměnných! Takže bacha na to, jaké proměnné si voláte a jaké zadáváte!

Kdo by si chtěl celkově věci z bioinformatiky propojit a pochopit doporučuji
→ https://slideplayer.com/slide/4837166/

Jste u konce! Děkuji za Váš čas!

Jana Schwarzerová
xschwa16@stud.feec.vutbr.cz