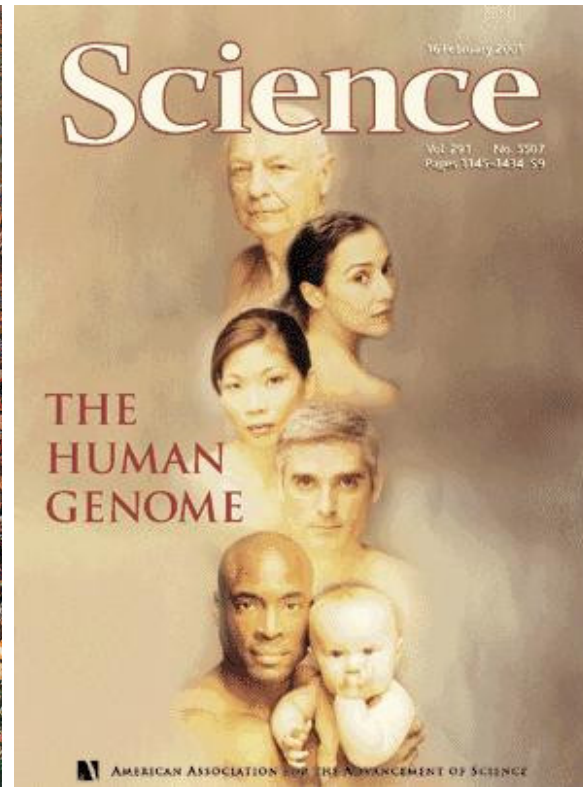


# Sekvenovanie a zostavovanie genómov (genome sequencing and assembly)

Broňa Brejová

22.9.2022



## Typický priebeh sekvenovania

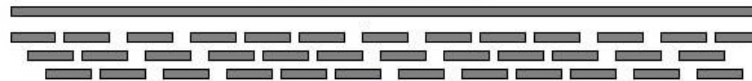
1. Chromozómy náhodne rozsekáme na menšie kúsky  
(napr. pomocou sonikácie)
2. Menšie kúsky namnožíme  
(napr. pomocou PCR, bakteriálneho klonovania a pod.)
3. Konce týchto kúskov osekvenujeme niektorou zo sekvenovacích technológií  
⇒ mnoho krátkych reťazcov, ktoré nazývame **čítania**
4. Čítania **výpočtovo zostavíme** späť do chromozómov

## Prehľad sekvenovacích technológií

Technológia	Dĺžka čítania	Chybovosť	Za deň	Cena za MB
<b>1. generácia</b>				
Sanger	do 1000 bp	< 1%	3 MB	\$4000
<b>2. (next) generácia (cca od 2004)</b>				
Illumina	250bp	< 0.1%	150 GB	\$0.03
<b>3. generácia (cca od 2018)</b>				
PacBio	cca 15 Kbp	≈ 10%	700 GB	\$0.02
PacBio HiFi	cca 15 Kbp	< 1%	70 GB	\$0.20
Oxford Nanopore	5-100+kbp	≈ 10%	50 GB	\$0.02

## Bioinformatický problém: Zostavenie genómu (sequence assembly)

- **Vstup:** krátke čítania sekvenovanej DNA
- **Cieľ:** zostaviť pôvodnú DNA
  - riadime sa zhodou v prekrývajúcich častiach čítaní
- Dôležité faktory:
  - **dĺžka genómu**
  - **dĺžka jednotlivých čítaní**
  - **pokrytie** (coverage) – koľko krát čítania pokrývajú genóm?



## Formulácia problému (jednoduchá, ale nerealistická)

### Najkratšie spoločné nadslovo (shortest common superstring)

**Úloha:** Daných je niekoľko reťazcov  $S_1, S_2, \dots, S_n$  (čítania), nájdite **najkratší** reťazec  $S$ , ktorý obsahuje **každý** vstupný reťazec  $S_i$  ako (súvislý) **podreťazec**.

Motivácia: čo najviac využiť prekryvy medzi čítaniami

### Príklad:

Vstup:  $S_1 = \text{GCCAAC}$ ,  $S_2 = \text{CCTGCC}$ ,  $S_3 = \text{ACCTTC}$

Výstup:  $S = \text{CCTGCCAACCTTC}$

(čítania spojené v poradí  $S_2, S_1, S_3$ )

## Najkratšie spoločné nadslovo

- **Problém je NP ťažký**

takže nepoznáme rýchly algoritmus, ktorý vždy nájde najlepšie riešenie

- **Jednoduchá heuristika:** opakovane nájdí dva reťazce, ktoré sa prekrývajú najviac a zlúč ich do jedného reťazca

- Príklad: CATATAT, TATATA, ATATATC

Optimum: CATATATATC, dĺžka 10

Heuristika: CATATATCTATATA, dĺžka 14

- V skutočnosti táto heuristika **aproximačný algoritmus:**

Nájdene riešenie je najviac  $3,5 \times$  horšie ako optimálne

T.j. je to 3,5-aproximačný algoritmus

(možno aj 2-aproximačný, otvorený problém)

- Existuje aj 2,5-aproximačný algoritmus

## Najkratšie spoločné nadslovo: Čo sme nezahrnuli do formulácie

- V sekvenovaní sa vyskytujú chyby
- Polymorfizmus
- Orientácia čítaní (vlákno, strand)
- Kontaminácia cudzou sekvenciou (napr. baktérie, v ktorých sa segmenty DNA klonovali), chiméry
- Viac chromozómov, neúplné pokrytie čítaniami
- Repetitívna sekvencia (sequence repeats, opakovania)

cca 50% ľudského genómu

Príklad: 10xTTAATA, 10xATATTA, 3xTTAGCT

TTAATATTAGCT?

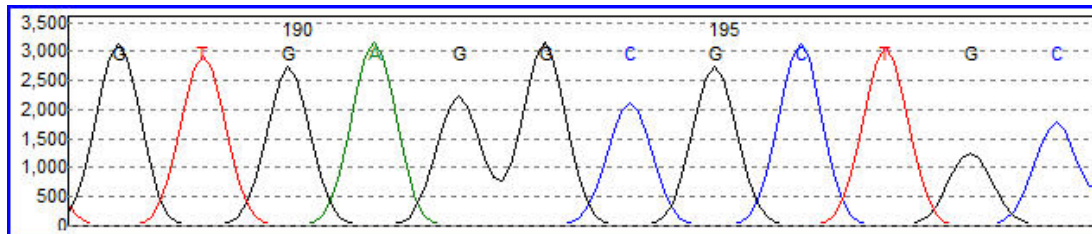
TTAATATTAATATTAATATTAATATTAGCT?

TTAATATTA + ATATTAGCT?

## Čo sme nezahrnuli do formulácie: kvalita báz

- K čítaniam máme väčšinou informáciu o **kvalite báz**  
Aká je pravdepodobnosť, že daná báza je správna?
- Báza s kvalitou  $q \Rightarrow$  pravdepodobnosť chyby  $10^{-q/10}$   
napr. báza s  $q = 40$  je správna s pr. 99.99%

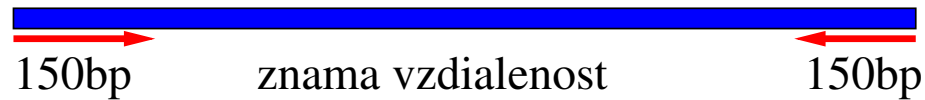
Príklad výsledku Sangervho sekvenovania (trace):





## Najkratšie spoločné nadslovo: Zľahčujúce faktory

**Prídavná informácia:** spárované čítania (pair-end reads)



**Zjednodušenie:** nemusíme spojiť všetko do jedného reťazca,  
spájame len časti spojené viacerými čítaniami

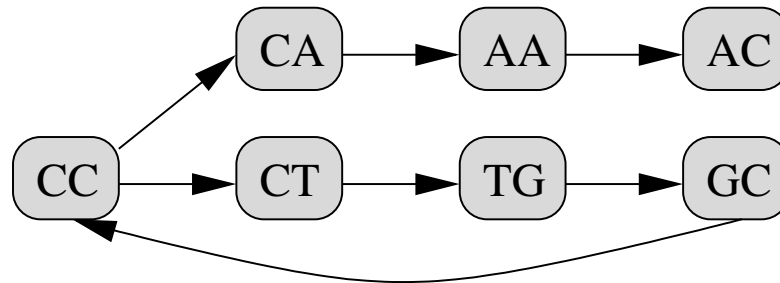
Konzervatívny prístup (radšej menej pospájať, ale nerobiť chyby)

## Najkratšie spoločné nadslovo: Zhrnutie

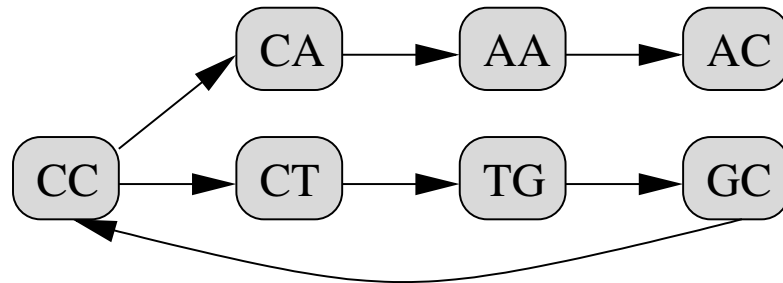
- Nerealistická formulácia, ťažký výpočtový problém
- Ale teoretický problém môže poskytnúť nejaký posun k pochopeniu skutočného problému
- Overlap-Layout-Consensus prístup  
motivovaný greedy algoritmami pre najkratšie spoločné nadslovo  
(budúci týždeň)

## Skladanie krátkych čítaní: de Bruijnov grafy

- Nasekajme čítania na (prekrývajúce sa) kúsky dĺžky  $k$
- Zostavme z nich **de Bruijnov graf**
  - **vrcholy**: podreťazce dĺžky  $k$  všetkých čítaní
  - **hrany**: nadväzujúce  $k$ -tice v rámci každého čítania (s prekryvom  $k - 1$ )
  - Graf je orientovaný (hrany majú smer)
- **Príklad**:  $k = 2$ , čítania: CCTGCC, GCCAAC



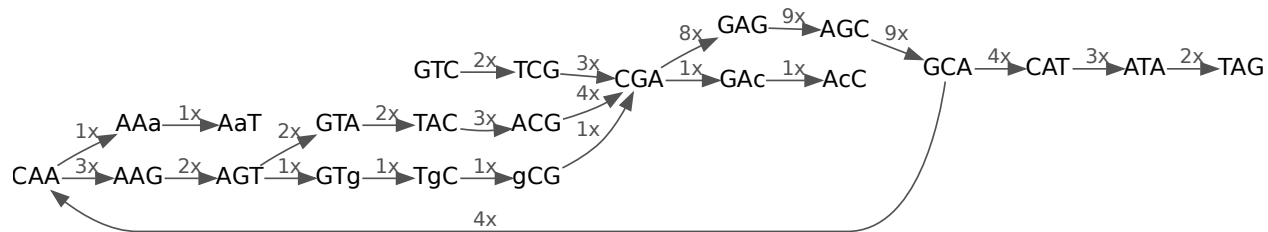
## Ako použiť de Bruijnové grafy?



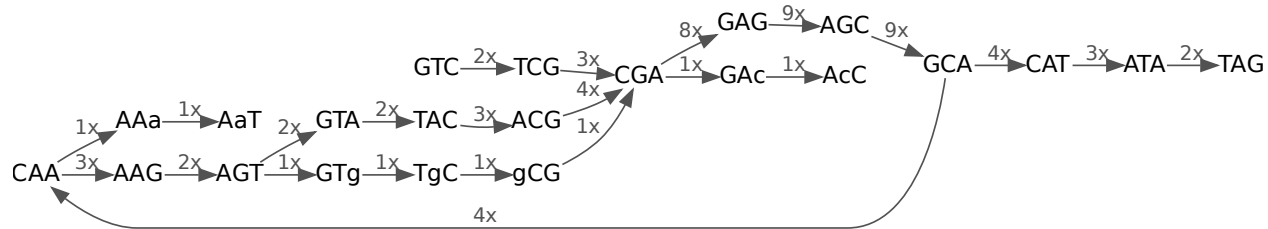
- jediný chromozóm a žiadne “nejednoznačné”  $k$ -tice  
⇒ zostavenie = **Eulerovská cesta**  
(cesta v grafe, ktorá použije každú hranu práve raz)
- Eulerovskú cestu možno nájsť v čase  $O(m + n)$
- v realistickom prípade:  
zostavenie genómu zodpovedá niekoľkým  
**pochôdkam v de Bruijnovom grafe** (nazývame **kontigy**),  
ktoré dohromady pokrývajú veľkú časť hrán

## Príklad: sada čítaní a zodpovedajúci deBruijnov graf

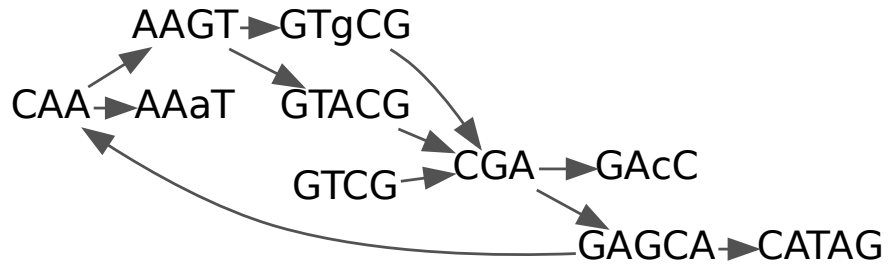
GTCGAGCAAGTACGAGCATAG  
 TCGAGCA AGCATAG  
 AGCAAaT AGCATAG  
 GTCGAcC GTACGAG  
 GTCGAGC TACGAGC  
 CGAGCAA ACGAGCA  
 AGTgCGA  
 CAAGTAC  
 GCAAGTA GAGCAT  
 GAGCAAG GAGCATA  
 TACGAGC



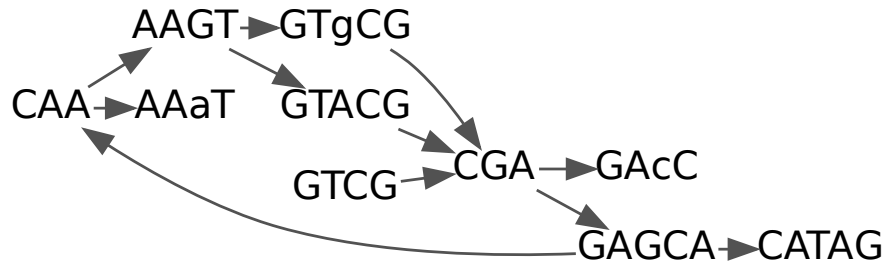
## Príklad: zjednodušovanie de Bruijnovho grafu



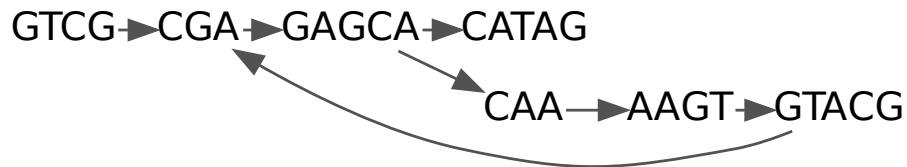
Spojíme jednoznačné cesty do vrcholov



## Príklad: odstraňovanie chýb z de Bruijnovho grafu



Odstránenie chýb (výbežkov a bublín s nízkym pokrytím)



Spájaním dostaneme 4 kontigy (pôv. GTCGAGCAAAGTACGAGCATAG)



## Typické výsledky zostavovania genómov

- Veľa **kratších kontigov**,  
niekedy spájané do väčších celkov (**scaffolds**) pomocou ďalšej informácie (napr. spárované čítania, čítania 3. generácie)
- Niektoré časti nemožno jednoznačne zostaviť z dôvodu **dlhých opakujúcich sa sekvencií**

**Príklad:** človek chr14, 88 Mbp, 70× pokrytie

Metóda	Počet kontigov	Chýb	N50 po korekcii
Velvet (základný de Bruijn)	>45000	4910	2.1 kbp
Velvet (scaffolding)	3565	9156	27 kbp
AllPaths-LG	225	45	4.7 Mbp

N50: kontigy s touto alebo dlhšou dĺžkou pokrývajú 50% genómu

korekcia: rozsekneme všetky zle spojené kontigy



## Zhrnutie

- Sekvenovanie genómu je zložitý proces, v ktorom hrá bioinformatika dôležitú úlohu
- Illumina nízka cena, krátke čítania
- Problém zostavovania genómu, najkratšie spoločné nadslovo
- Praktické riešenie pre krátke čítania: de Bruijnove grafy
- V zostavenej sekvencii môžu byť chyby, medzery, viaceré kontigy
- Na budúce: ako sa vysporiadať s dlhými čítaniami 3. generácie?
- Pokrytie genómu a veľkosť čítania hrajú najdôležitejšiu úlohu pri tom, ako fragmentovaný bude výsledok:
  - pre Sanger: 7-10× pokrytie
  - pre NGS: 40-70× pokrytie
  - pre 3. generáciu: 30× pokrytie

## História sekvenovania genómov

- 1976 MS2 (RNA vírus) 40 kB
- 1988 projekt sekvenovania ľudského genómu (15 rokov)
- 1995 baktéria *H. influenzae* 2 MB, shotgun (TIGR)
- 1996 *S. cerevisiae* 10 MB, BAC-by-BAC (Belgicko, Británia)
- 1998 *C. elegans* 100 MB, BAC-by-BAC (Wellcome Trust)
- 1998 Celera: ľudský genóm do troch rokov!
- 2000 *D. melanogaster* 180 MB, shotgun (Celera, Berkeley)
- 2001 2x ľudský genóm 3 GB (NIH, Celera)
- po 2001 Myš, potkan, kura, šimpanz, pes, . . .
- 2007 Watsonov a Venterov genóm (454)
- 2012 1000 ľudských genómov
- čoskoro 10k genómov stavovcov, sekvenovanie ako diagnostický nástroj
- 2021 3,5 milióna genómov SARS-CoV-2
- 2022 Naozaj dokončený ľudský genóm (telomere to telomere)