

## Oznamy

- DÚ3 je zverejnená, odovzdajte do 13.12.
- Budúci štvrtok 8.12.:
  - prednáška aj cvičenia bežia normálne
- Štvrtok 15.12.:
  - nepovinné prezentácie journal clubu v čase prednášky
  - cvičenia pre informatikov aj biológov
- Termíny na konci semestra
  - DÚ3 utorok 13.12., správy zo journal clubu piatok 16.12.
- Budúci štvrtok dohodneme:
  - či chcete prezentovať journal club (dohodnite sa v skupinách)
  - termín skúšky pre informatikov (doneste si termíny iných skúšok)

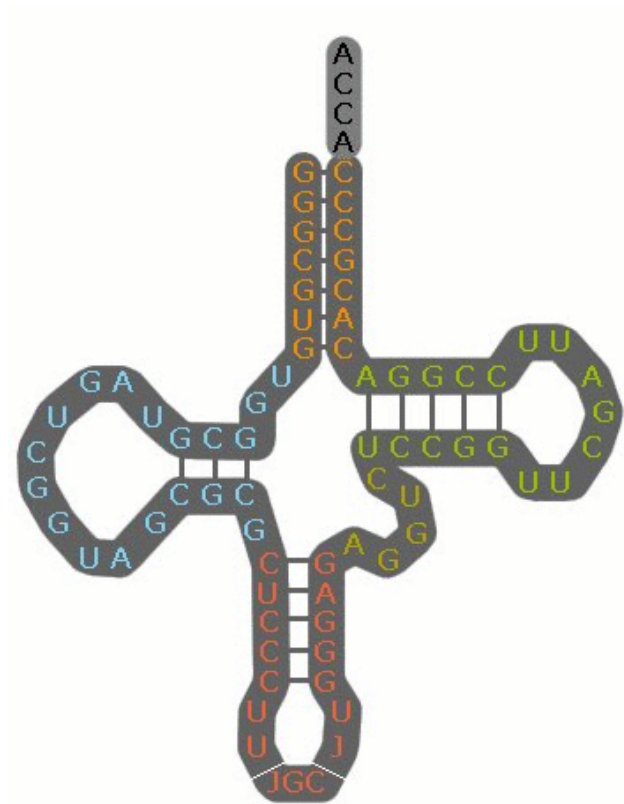
## Správa zo journal clubu

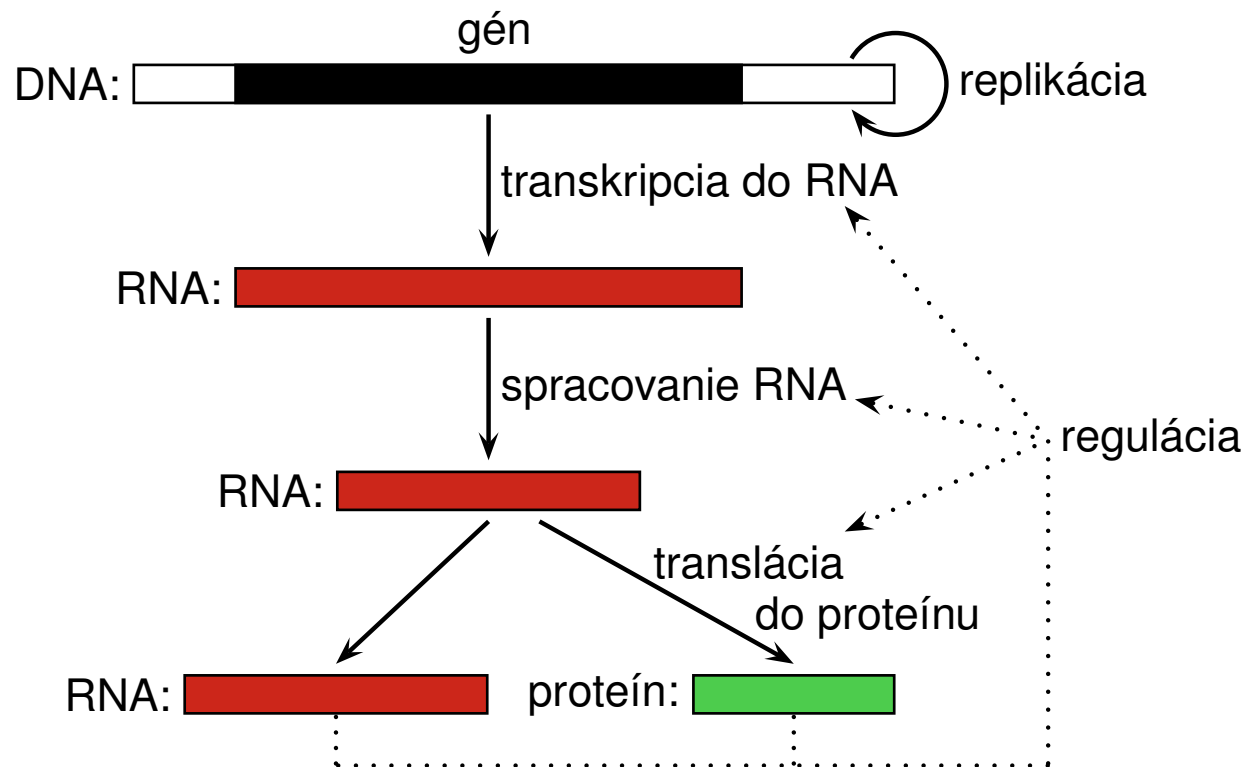
- Vlastnými slovami hlavné metódy a výsledky článku
- Pochopiteľná pre študentov tohto predmetu (inf aj bio)
- Netreba pokryť všetko a naopak, môžete využiť aj iné zdroje
- Skúste vložiť vlastný pohľad na tému, nepostupujte striktne podľa textu článku
- Rozsah cca 1-2 strany na osobu, jeden ucelený text
- V správe vymenujte členov skupiny, ktorí sa podieľali na jej spísaní, dostanú rovnako bodov
- Pdf odovzdať cez Moodle (stačí 1 za skupinu)

# RNA

Tomáš Vinař

1.12.2022





## Vlastnosti RNA

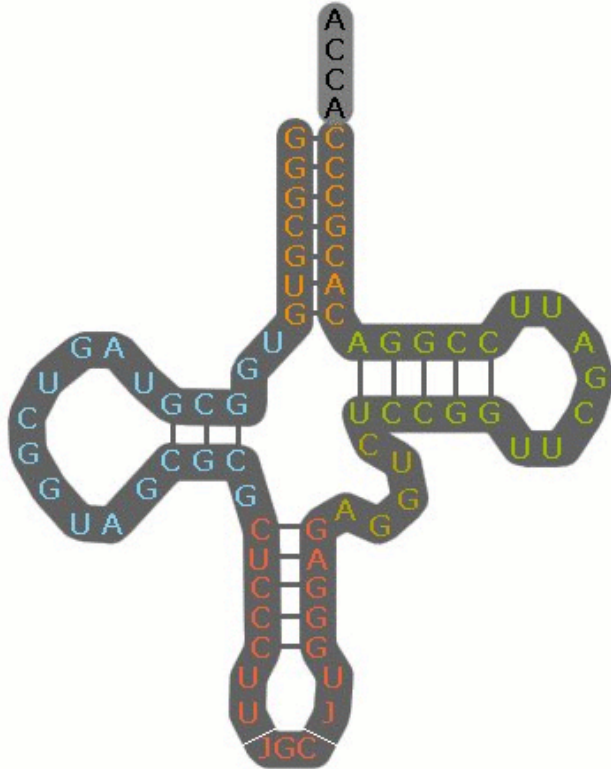
### Ako sa líši od DNA?

- obsahuje ribózu namiesto deoxyribózy
- obsahuje uracil namiesto tymínu (bázy A,C,G,U)
- jednovláknové reťazce, zvyčajne kratšie
- zložitá sekundárna štruktúra: spárované komplementárne úseky
- okrem párov A-U, C-G aj nekanonické páry (napr. G-U)
- rôzne funkcie v bunke:
  - centrálna úloha pri expresii génov (mediátorová, transferová, ribozómová RNA),
  - regulácia expresie,
  - katalytické funkcie,
  - prenos genetickej informácie pre RNA vírusy

## Štruktúra RNA

### Príklad: transferová RNA (transfer RNA)

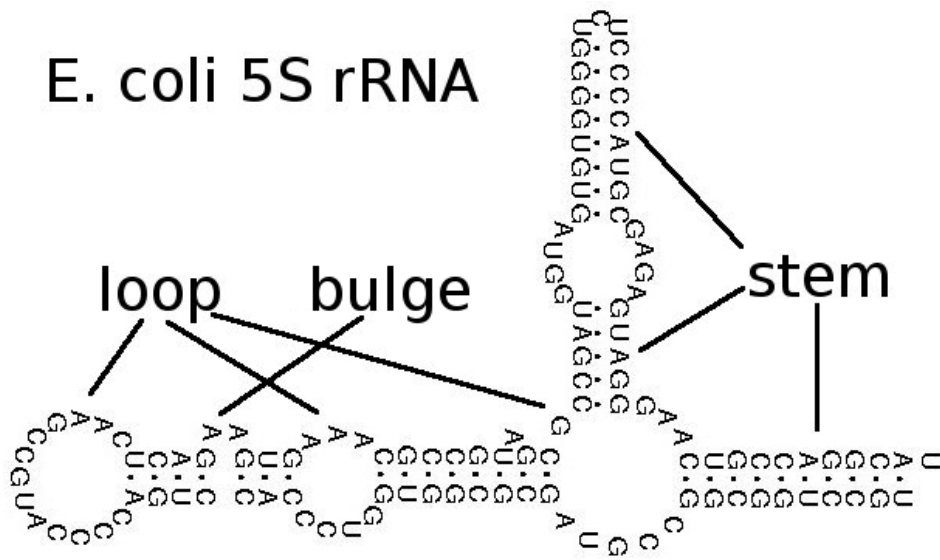
Sekundárna štruktúra  
(secondary structure):  
páry nukleotidov



Terciárna štruktúra  
(tertiary structure):  
3D súradnice



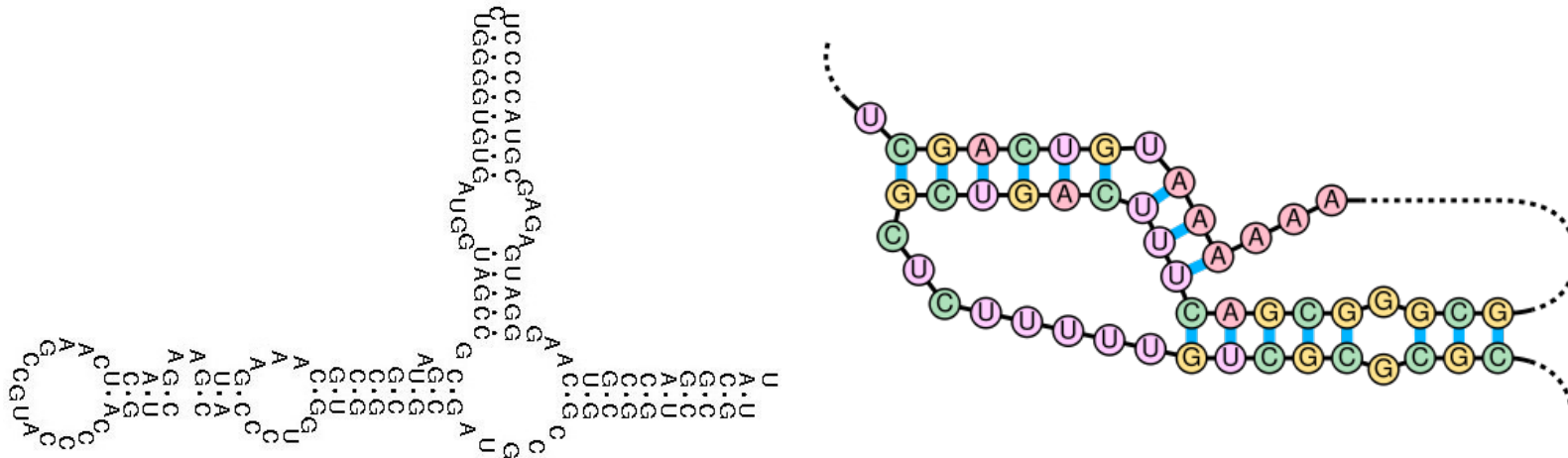
## Sekundárna štruktúra RNA



### Zápis sekundárnej štruktúry pomocou zátvoriek:

((( ((( ((( ((( (..... ((( ) .)) . (( ) ) ...))) ) ) ) ) ) ) ) ) ) ) .  
 UGCCUGGCGGCCGUAGCG...UAGCGCC...GGGAACUGCCAGGCAU

## Dobre uzatvorkované výrazy vs pseudouzly



Príklad **vľavo**: spárované bázy tvoria **dobre uzatvorkovaný výraz**:

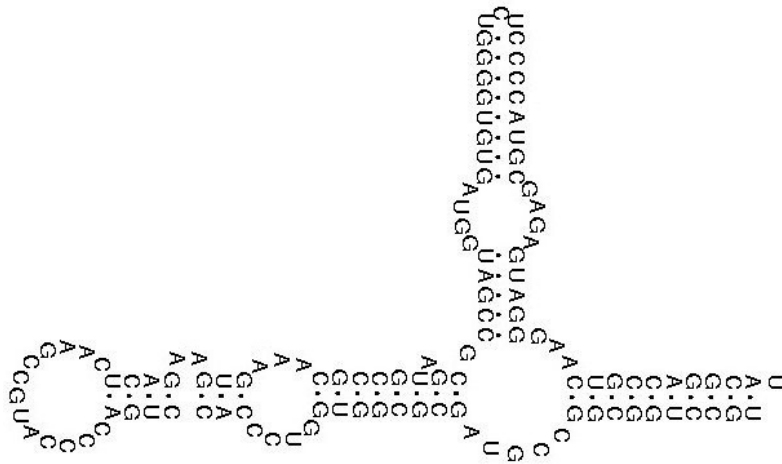
((((( (((((((.....((( ) .)) . (( ) ) ...))))) ) ) ) ) ) ) ) .  
 UGCCUGGCGGCCGUAGCG . . . UAGCGCC . . . GGGAACUGCCAGGCAU

Príklad **vpravo**: **pseudouzol** (výnimka z dobrého uzatvorkovania)

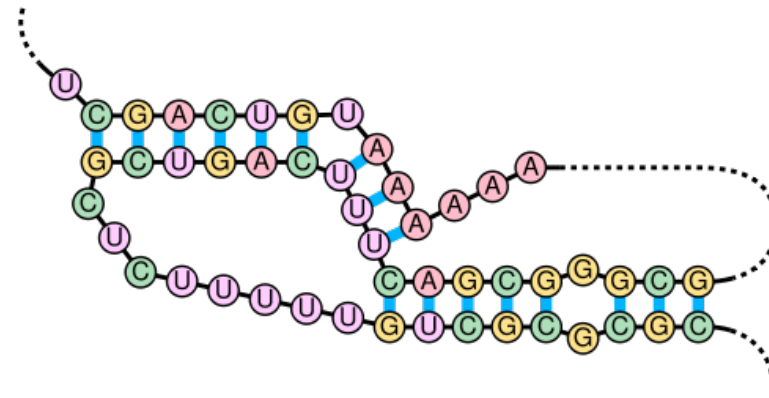
. (((((( ( . (((... [[ [. [[[[[ ] ) ) ) ) ) ) ) ) ) ) ) ) ) ) . . ] ] ] ] ] . ] ] ]  
 UCGACUGUAAAAAGCGGGCGACUUUCAGUCGC . . . UGUCGCGCGC



## Dobre uzátvorkované výrazy vs pseudouzly



bez pseudouzlu

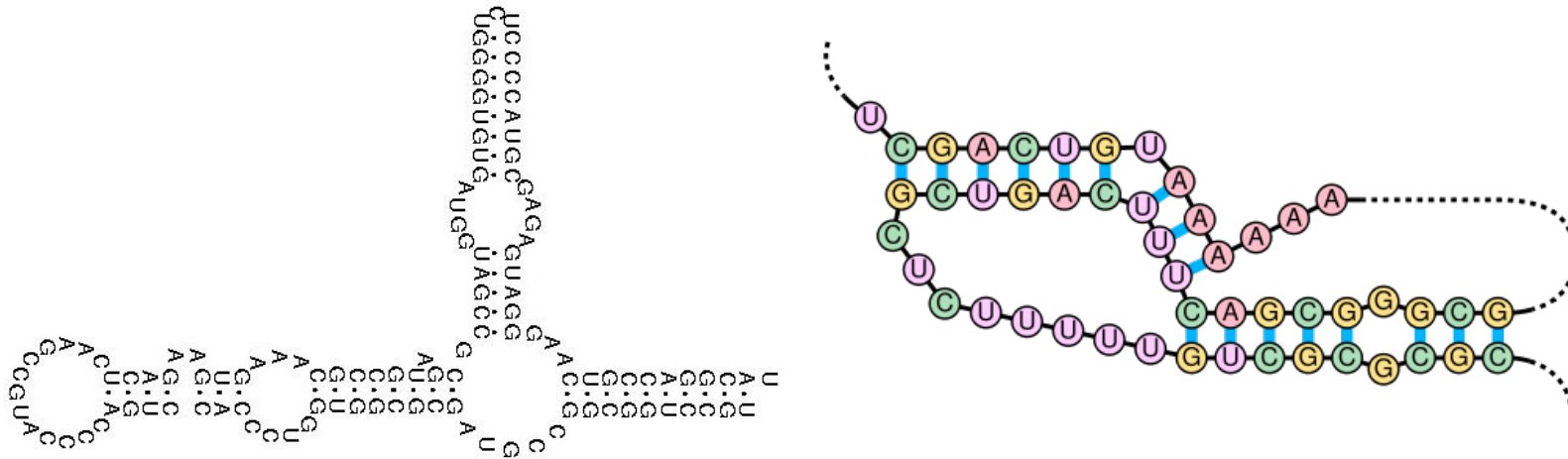


pseudouzol

Mnohé algoritmy na prácu so sekundárnou štruktúrou ignorujú pseudouzly.

Zhruba 1.4% RNA nukleotidových párov v pseudouzloch.

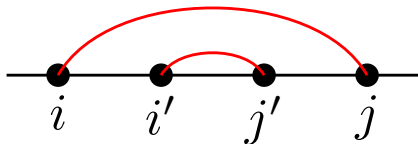
## Dobre uzatvorkované výrazy vs pseudouzly



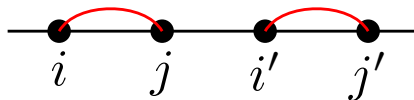
### Matematické vyjadrenie štruktúry bez pseudouzlov:

Ak máme páry medzi pozíciami  $i$  a  $j$  a  $i'$  a  $j'$  pričom  $i < i'$ , tak buď  $i < i' < j' < j$  alebo  $i < j < i' < j'$ .

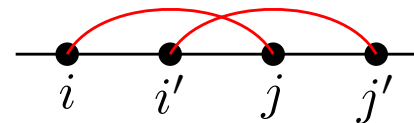
dobre:



dobre:



zle:



## Problém: určovanie štruktúry RNA

**Vstup:** RNA sekvencia

**Cieľ:** nájsť spárované bázy

**Veľmi zjednodušená formulácia:** nájsť dobre uzátvorkované spárovanie s najväčším počtom komplementárnych párov A-U, C-G.

### Príklad:

Vstup: ((.(((( ))) (((.))))))

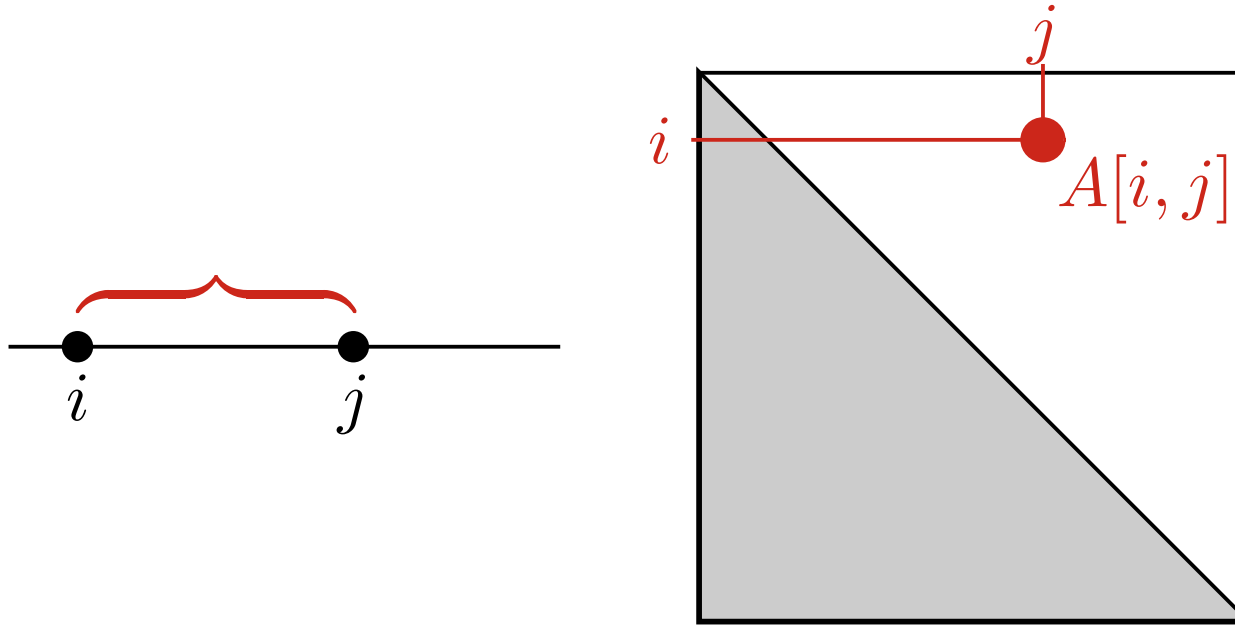
Výstup: GAACACAUGUAAAAUUUGUC

## Nussinovej algoritmus

### Dynamické programovanie:

Majme RNA  $x_1, \dots, x_n$ .

Nech  $A[i, j]$  je maximálny počet párov v podreťazci  $x_i, x_{i+1}, \dots, x_j$ .



## Nussinovej algoritmus

### Dynamické programovanie:

Majme RNA  $x_1, \dots, x_n$ .

Nech  $A[i, j]$  je maximálny počet párov v podreťazci  $x_i, x_{i+1}, \dots, x_j$ .

### Rekurencia:

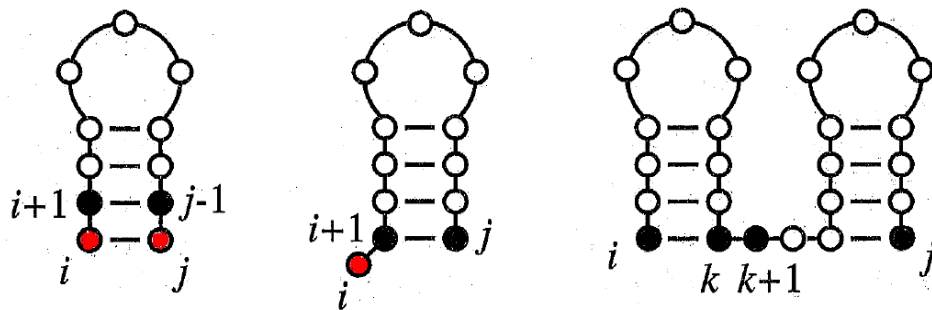
Podreťazce dĺžky 1: žiadne páry  $A[i, i] = 0$

Dlhšie podreťazce: 3 prípady

–  $x_i$  a  $x_j$  sú pár:  $A[i, j] = A[i + 1, j - 1] + 1$

–  $x_i$  je nespárované:  $A[i, j] = A[i + 1, j]$

–  $x_i$  je pár s  $x_k$  pre  $i < k < j$ :  $A[i, j] = A[i, k] + A[k + 1, j]$

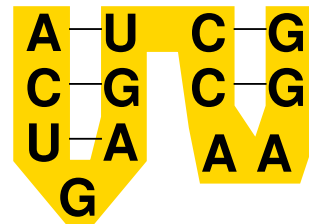


**Rekurencia:**  $A[i, j] = \max \begin{cases} A[i + 1, j - 1] + c(x_i, x_j), \\ A[i + 1, j], \\ \max_{k=i+1 \dots j-1} \{A[i, k] + A[k + 1, j]\} \end{cases}$

	A	C	U	G	A	G	U	C	C	A	A	G	G
A	0	0	1	1	1	2	3	3	3	3	3	4	5
C		0	0	1	1	2	2	2	3	3	3	4	4
U			0	0	1	1	1	2	2	3	3	3	3
G				0	0	0	1	2	2	2	2	3	3
A					0	0	1	1	1	1	1	2	3
G						0	0	1	1	1	1	2	2
U							0	0	0	1	1	1	2
C								0	0	0	0	1	2
C									0	0	0	1	1
A										0	0	0	0
A											0	0	0
G												0	0
G													0

$c(x_i, x_j) = \begin{cases} 1 & \text{ak } x_i - x_j \text{ môže byť pár} \\ 0 & \text{inak} \end{cases}$

$A[i, j] = 0$  pre  $i \geq j$



**Zložitosť:**

$O(n^3)$  čas

$O(n^2)$  pamäť

## Štruktúra s minimálnou voľnou energiou (MFE folding)

Realistickejšia formulácia problému určovania sekundárnej štruktúry RNA.

**Predpoklad:** molekula v rovnovážnom stave

s minimálnou Gibbsovou voľnou energiou (Gibbs free energy).

Energie pre niektoré sekvencie experimentálne zmerané.

**Nearest neighbor model:** sada parametrov, energie pre dvojice susedných párov v helixoch, dĺžky slučiek atď.

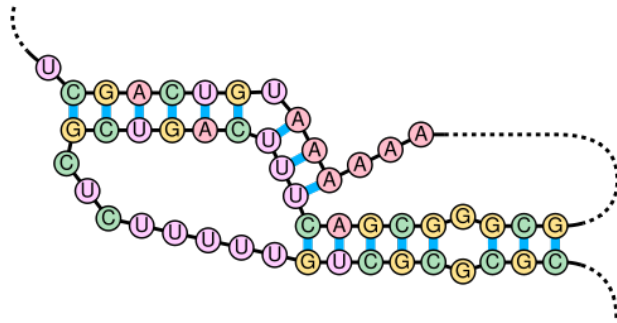
Odvođené z nameraných dát.

### Príklad:

		Y:	A	C	G	U		
5'	CX	3'	-----					
3'	GY	5'	X:A		.	.	.	-2.1
			C		.	.	-3.3	.
			G		.	-2.4	.	-1.4
			U		-2.1	.	-2.1	.

Štruktúra s minimálnou energiou sa dá nájsť podobným (ale zložitejším) dyn. programovaním (Zuker and P. Stiegler, 1981)

## Algoritmy dovoľujúce pseudouzly



Vo všeobecnosti NP-ťažký problém (Lyngso and Pedersen 2000).

Pomalé dyn. programovanie  $O(n^4) - O(n^6)$  nájde niektoré typy pseudouzlovov (Rivas and Eddy 1999)

Tiež môžeme použiť heuristiky (opakované vytváranie silných helixov) alebo celočíselné lineárne programovanie (posledné cvičenia pre informatikov)



## **Pravdepodobnostné modely na predikciu štruktúry**

Chceme: model, ktorý generuje dvojice sekvencia a sek. štruktúra

Použitie: pre danú sekvenciu nájsť najpravdepodobnejšiu štruktúru

HMM nevhodné: závislosti medzi vzdialenými spárovanými bázami.

## **Stochastická bezkontextová gramatika, stochastic context free grammar (SCFG):**

Rozšírenie bezkontextových gramatík

Pravidlám pridáme pravdepodobnosti

## Stochastické bezkontextové gramatiky (SCFG)

neterminály (veľké písmená) podobné na stavy v HMM,  
terminály (malé písmená) reprezentujú nukleotidy.

Pravidlá prepisujú neterminál na reťazec terminálov a neterminálov.  
Každé pravidlo má pravdepodobnosť.

**Príklad:** jeden neterminál, 14 pravidiel ( $\epsilon$  =prázdny reťazec)

$$\begin{array}{cccc}
 0.1 & 0.1 & 0.1 & 0.1 \\
 \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} \\
 S \rightarrow aSu & | & uSa & | & cSg & | & gSc & | \\
 0.05 & 0.05 & 0.05 & 0.05 & 0.05 & 0.05 & 0.05 & 0.05 & 0.1 & 0.1 \\
 \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} \\
 aS & | & cS & | & gS & | & uS & | & Sa & | & Sc & | & Sg & | & Su & | & SS & | & \epsilon
 \end{array}$$

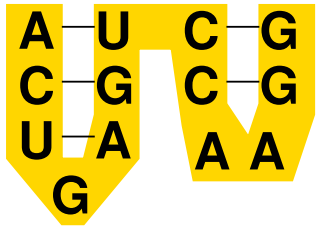
V každom kroku zvoľ jeden (napr. najľavejší) neterminál,  
prepíš ho náhodne zvoleným pravidlom:

$$\begin{aligned}
 S &\rightarrow SS \rightarrow aSuS \rightarrow acSguS \rightarrow acuSaguS \rightarrow acugSaguS \rightarrow \\
 &acugaguS \rightarrow acugagucSg \rightarrow acugaguccSgg \rightarrow acugaguccSagg \rightarrow \\
 &acugaguccaSagg \rightarrow acugaguccaagg
 \end{aligned}$$

## Stochastické bezkontextové gramatiky

$S \rightarrow aSu|uSa|cSg|gSc|aS|cS|gS|uS|Sa|Sc|Sg|Su|SS|\epsilon$

$S \rightarrow SS \rightarrow aSuS \rightarrow acSguS \rightarrow acuSaguS \rightarrow acugSaguS \rightarrow$   
 $acugaguS \rightarrow acugagucSg \rightarrow acugagucgScg \rightarrow acugagucgSacg \rightarrow$   
 $acugagucgaSacg \rightarrow acugagucgaacg$



Bázy vygenerované v jednom kroku sú spárované.

**Úloha:** Nájdi najpravdepodobnejšie odvodenie danej RNA

⇒ určuje sekundárnu štruktúru

**Riešenie:** Dynamické programovanie (CYK algoritmus),  $O(n^3)$

**Trénovanie parametrov:** zo známych RNA štruktúr

## Gramatiky vs. minimalizácia energie

### Výhody gramatík:

- možno automaticky trénovať, netreba náročné experimenty,
- rozšíriteľné na modely viacerých sekvencií.

### Nevýhody gramatík:

- jednoduché gramatiky nevystihujú všetky aspekty problému,
- nižšia presnosť ako minimalizácia energie.

## Evolúcia RNA sekvencií

Často vidíme koreláciu medzi mutáciami v spárovaných bázach.  
Např. C sa zmení na A, spárované G sa súčasne zmení na U

**Príklad:** niekoľko sekvencií z D ramena tRNA

```
(((((.....))))  
GCUCAGCC .CGGG . . .AGAGC  
GCCUAGCC .UGGUCA .AGGGC  
GUCUAGC . . .GGA . . .AGGAU  
GAGCAGUU .CGGU . . .AGCUC  
GUUCAAUC . .GGU . . .AGAAC
```

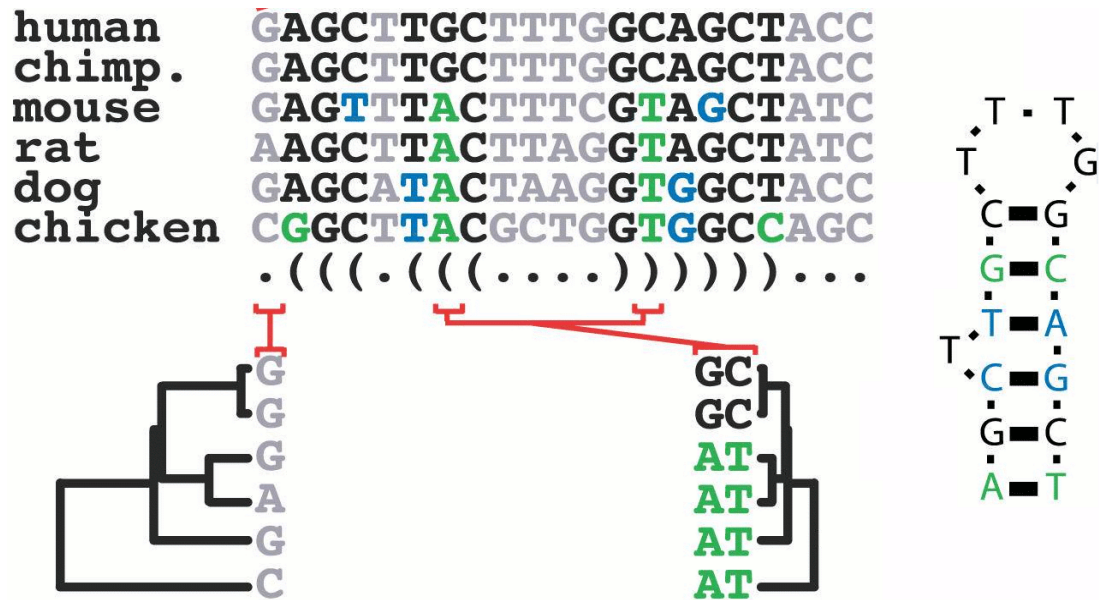
**Úloha:** daných je niekoľko (zarovnaných) sekvencií RNA  
nájdite ich spoločnú RNA štruktúru

(korelácie medzi spárovanými bázami potvrdzujú správnosť štruktúry)

## Hľadanie spoločnej štruktúry pre viacero sekvencií

### Phylo-SCFG:

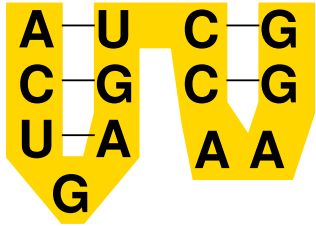
- namiesto jednotlivých báz emituje stĺpce zarovnaní podľa fylogenetického stromu.
- nespárované bázy emituje bežnou substitučnou maticou,
- spárované bázy substitučnou maticou dvojíc ( $16 \times 16$ ).



## Problém: hľadanie známych typov RNA génov v genóme

- Databáza Rfam: štruktúry pre >4000 rodín RNA génov
- Pre každú rodinu zarovnanie a pravdepodobnostný model
- Pre RNA kovariančné modely (covariance model, CM):  
špeciálny typ SCFG
- Podobná myšlienka ako profilové HMM pre proteínové rodiny

## Kovariančný model



$$\begin{array}{lll}
 S \rightarrow B_1 & P_1 \rightarrow aP_2u & P_4 \rightarrow cP_5g \\
 B_1 \rightarrow P_1P_4 & P_2 \rightarrow cP_3g & P_5 \rightarrow gL_2c \\
 & P_3 \rightarrow uL_1a & L_2 \rightarrow aL_3 \\
 & L_1 \rightarrow gE_1 & L_3 \rightarrow aE_2 \\
 & E_1 \rightarrow \epsilon & E_2 \rightarrow \epsilon
 \end{array}$$

- $S$  =start,  $E_i$  =end  
 $P_i$  =pár,  $L_i$  =nespárovaná báza vľavo,  $R_i$  =nespárovaná báza vpravo  
 ďalšie neterminály modelujú indely.
- terminály (bázy) sa emitujú s pravdepodobnosťami **podľa príslušného stĺpca zarovnaní**

$$\text{Např. } P_1 \rightarrow \overbrace{aP_2u}^{0.2} \mid \overbrace{uP_2a}^{0.2} \mid \overbrace{cP_2g}^{0.4} \mid \overbrace{cP_2u}^{0.1}$$

- veľkosť gramatiky úmerná dĺžke modelovanej RNA rodiny



## Kovariančný model

### Použitie:

hľadať výskyty génu v DNA (lokálne zarovnanie),  
nájsť štruktúru nového génu z tej istej rodiny (globálne zarovnanie).

**Dynamické programovanie:** čas  $O(MND^2)$ ,

$M$  = počet neterminálov v gramatike, úmerný dĺžke zarovnania,

$N$  = dĺžka DNA sekvencie,

$D$  = max. dĺžka RNA génu v DNA (úmerná  $M$ ).

### Zrýchlenie:

nájsť sľubné úseky podobné na sekvencie v RNA rodine

(iba na základe podobnosti sekvencií)

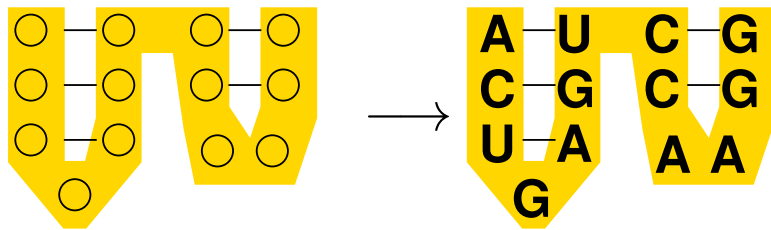
aplikuj CM iba na sľubné úseky

## Problém: dizajn RNA

Daná RNA sekundárna štruktúra (párovanie).

Nájdí sekvenciu, pre ktorú je táto štruktúra optimálna.

Nie je známy efektívny algoritmus, heuristiky často nájdu sekvenciu pomerne rýchlo.



**Použitie:** skúmanie možných RNA štruktúr, vývoj liekov (ribozymes, riboswitches), RNA pre laboratórne techniky, RNA nanoštruktúry

## Zhrnutie

- Určovanie sekundárnej štruktúry RNA:  
minimalizácia energie, pravdepodobnostné SCFG
- Lepšie výsledky, keď použijeme zarovnanie viacerých sekvencií (PhyloSCFG)
- Známe rodiny reprezentujeme pomocou kovariančných modelov  
v nových sekvenciách hľadáme výskytu rodín z databázy Rfam
- Väčšina problémov sa dá riešiť dynamickým programovaním, ktoré je pomerne pomalé a ignoruje pseudouzly.
- Ďalšie zaujímavé problémy: napr. dizajn RNA štruktúr