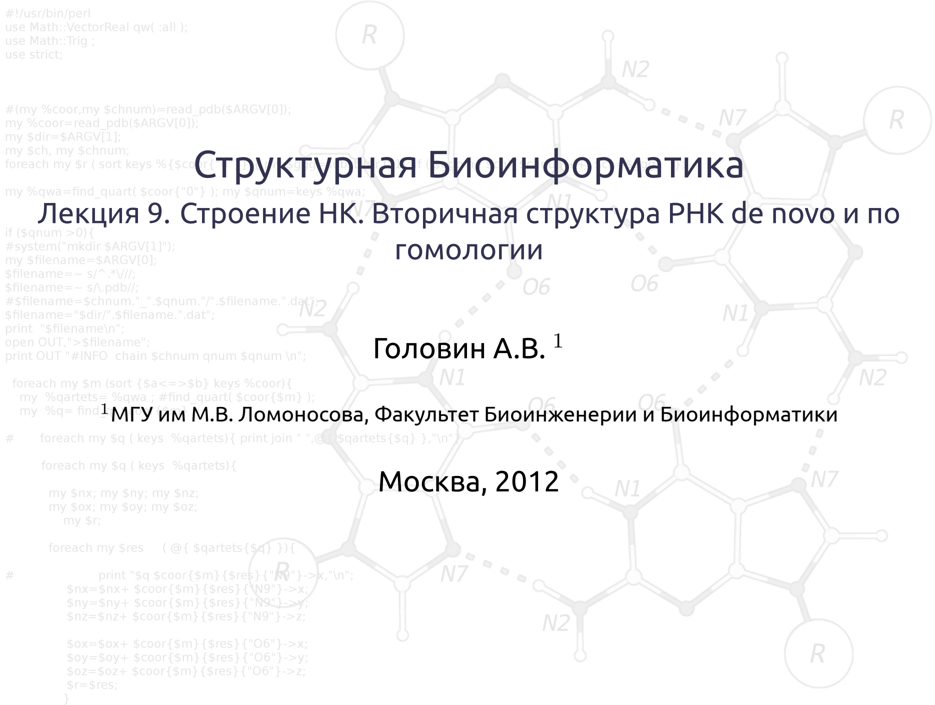


Структурная Биоинформатика

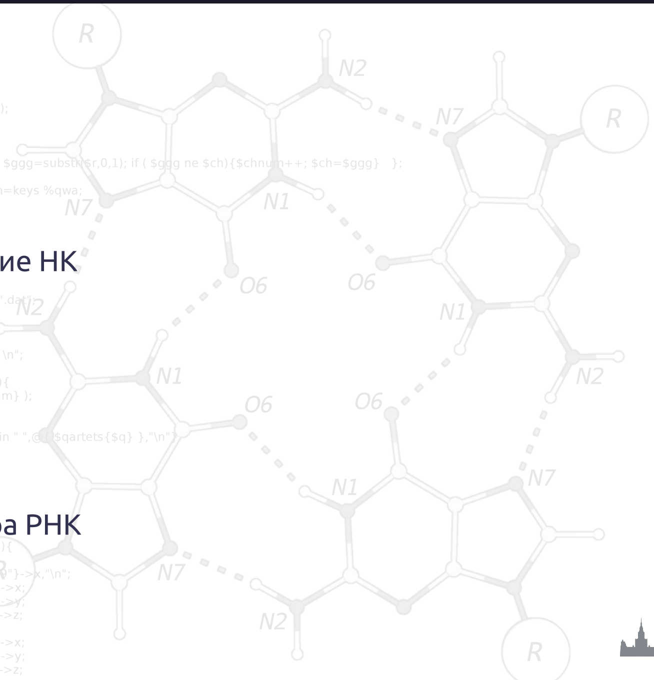
Лекция 9. Строение НК. Вторичная структура РНК de novo и по гомологии

Головин А.В. [1]

[1] МГУ им М.В. Ломоносова, Факультет Биоинженерии и Биоинформатики

Москва, 2012

Содержание



Введение

Химическое строение НК

3D Структура ДНК

Структура РНК

Вторичная структура РНК

Нуклеиновые кислоты.

Нуклеиновые кислоты (НК) – это высокомолекулярные линейные полярные биополимеры, **полинуклеотиды**, которые построены из **нуклеотидных остатков.**

2 типа нуклеиновых кислот:

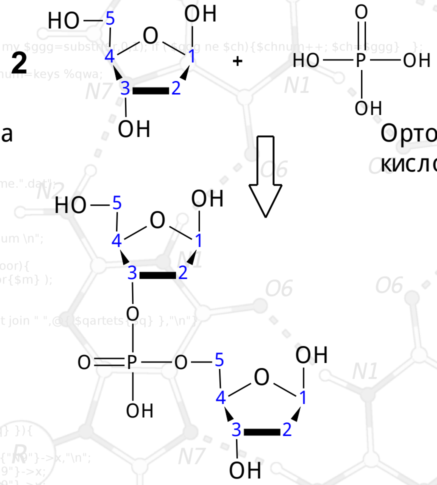
- дезоксирибонуклеиновая кислота – ДНК
- рибонуклеиновая кислота – РНК

ДНК

Расшифровка аббревиатуры ДНК:

- Д**Н**К – открытие и выделение **"нуклеина"** из ядер (нуклеус) лейкоцитов Ф. Мишером 1869 г.
- ДН**К** – линейный сополимер на основе ортофосфорной **кислоты**.
- **Д**НК – линейный сополимер ортофосфорной кислоты и **дезоксирибозы**

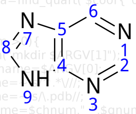
Дезоксирибоза и ортофосфорная кислота



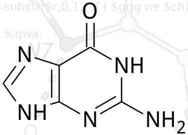
Дезоксирибоза

Ортофосфорная кислота

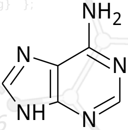
Основания



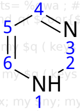
Purine



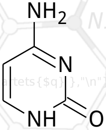
Guanine



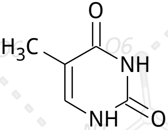
Adenine



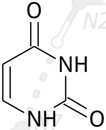
Pyrimidin



Cytosine



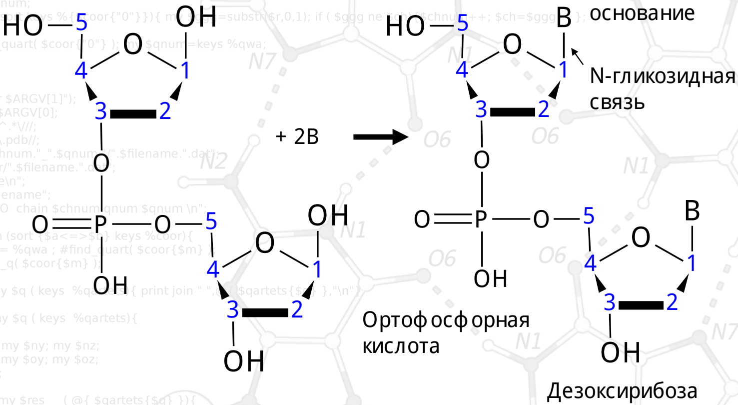
Thymine



Uracil

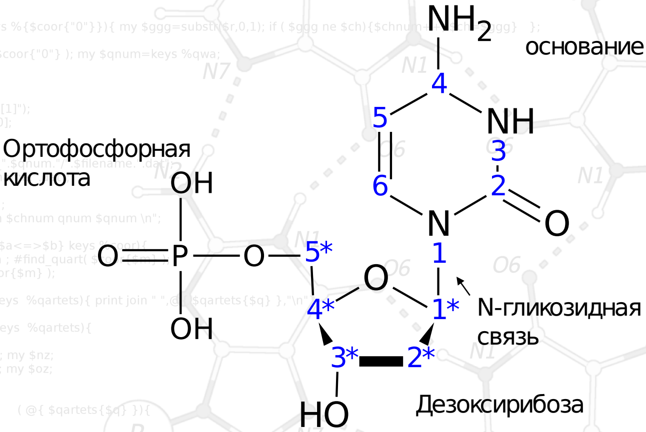
```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
use Math::Trig;
use strict;

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}}){ my $ggg=substr($r,0,1); if ( $ggg ne $ch){$chnum++; $ch=$ggg}   };

my %qwa=find_quart( $coor{"0"} ); my $qnum=keys %qwa;

if ($qnum>0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename=~s/^.*\///;
$filename=~s/\.pdb//;
#$filename=$chnum."_".$qnum."/".$filename.".dat";
$filename="$[illegible]".$filename.".dat";
print  "$filename\n";
open OUT,">$filename";
print OUT "#INFO  chain $chnum qnum $qnum \n";

 foreach my $m (sort {$a<=>$b} keys %coor){
   my  %qartets= %qwa ; #find_quart( $coor{$m} );
   my  %q= find_q( $coor{$m} );

#       foreach my $q ( keys  %qartets){ print join " ",@{ $qartets{$q} },"\n"}

        foreach my $q ( keys  %qartets){

          my $nx; my $ny; my $nz;
          my $ox; my $oy; my $oz;
            my [illegible]

          foreach my $res     ( @{ $qartets{$q} }){

#                     print "$q $coor{$m}{$res}{"N9"}->x,"\n";
              $nx=$nx+ $coor{$m}{$res}{"N9"}->x;
              $ny=$ny+ $coor{$m}{$res}{"N9"}->y;
              $nz=$nz+ $coor{$m}{$res}{"N9"}->z;

              $ox=$ox+ $coor{$m}{$res}{"O6"}->x;
              $oy=$oy+ $coor{$m}{$res}{"O6"}->y;
              $oz=$oz+ $coor{$m}{$res}{"O6"}->z;
```

Основания и остов



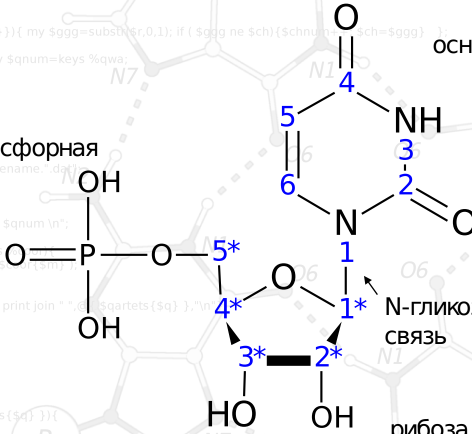
```
#!/usr/bin/perl
use Math::VectorReal qw( :all );

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}}){ my $ggg=substr($r,0,1); if ( $ggg ne $ch) {$chnum++; $ch=$ggg; };

my %qwa=find_quart( $coor{"0"} ); my $qnum=keys %qwa;

if ($qnum >0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename=~ s/^.*\///;
$filename=~ s/\.pdb//;
#$filename=$chnum."_".$qnum."".$filename.".dat";
$filename="$dir/".$filename.".dat";
print  "$filename\n";
open OUT,">$filename";
print OUT "#INFO  chain $chnum qnum $qnum \n";

 foreach my $m (sort {$a<=>$b} keys %coor){
   my  %qartets= %qwa ; #find_quart( $coor{$m} );
   my  %q= find_q( $coor{$m} );

#       foreach my $q ( keys  %qartets){ print join " ",(@{ $qartets{$q} }),"\n"}

        foreach my $q ( keys  %qartets){

          my $nx; my $ny; my $nz;
          my $ox; my $oy; my $oz;
             my $r;

          foreach my $res     ( @{ $qartets{$q} }){

#                  print "$q $coor{$m}{$res}{"N9"}->x,"\n";
            $nx=$nx+ $coor{$m}{$res}{"N9"}->x;
            $ny=$ny+ $coor{$m}{$res}{"N9"}->y;
            $nz=$nz+ $coor{$m}{$res}{"N9"}->z;

            $ox=$ox+ $coor{$m}{$res}{"O6"}->x;
            $oy=$oy+ $coor{$m}{$res}{"O6"}->y;
            $oz=$oz+ $coor{$m}{$res}{"O6"}->z;
```

Нуклеотид ДНК



Нуклеотид РНК

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
use Math::Trig;
use strict;

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}}){ my $ggg=substr($r,0,1); if ( $ggg ne $ch){$chnum++; $ch=$ggg }  };

my %qwa=find_quart( $coor{"0"} ); my $qnum=keys %qwa;


if ($qnum >0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename=~ s/^.*\///;
$filename=~ s/\.pdb//;
#$filename=$chnum."_".$qnum."/".$filename.".dat";
$filename="$dir/".$filename.".dat";
print  "$filename\n";
open OUT,">$filename";
print OUT "#INFO  chain $chnum qnum $qnum \n";

  foreach my $m (sort {$a<=>$b} keys %coor){
    my  %qartets= %qwa ; #find_quart( $coor{$m} );
    my  %q= find_q( $coor{$m} );

#       foreach my $q ( keys  %qartets){ print join " ",@{ $qartets{$q} },"\n"; }

        foreach my $q ( keys  %qartets){

          my $nx; my $ny; my $nz;
          my $ox; my $oy; my $oz;
              my $r;

          foreach my $res     ( @{ $qartets{$q} }){
#                       print "$q $coor{$m}{$res}{"N9"}->x,"\n";
               $nx=$nx+ $coor{$m}{$res}{"N9"}->x;
               $ny=$ny+ $coor{$m}{$res}{"N9"}->y;
               $nz=$nz+ $coor{$m}{$res}{"N9"}->z;

               $ox=$ox+ $coor{$m}{$res}{"O6"}->x;
               $oy=$oy+ $coor{$m}{$res}{"O6"}->y;
               $oz=$oz+ $coor{$m}{$res}{"O6"}->z;
```



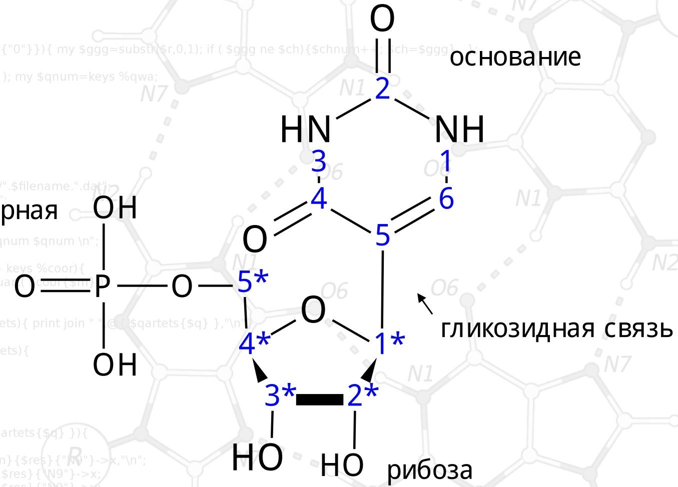
основание

Ортофосфорная кислота

N-гликозидная связь

рибоза

Псевдоуридин



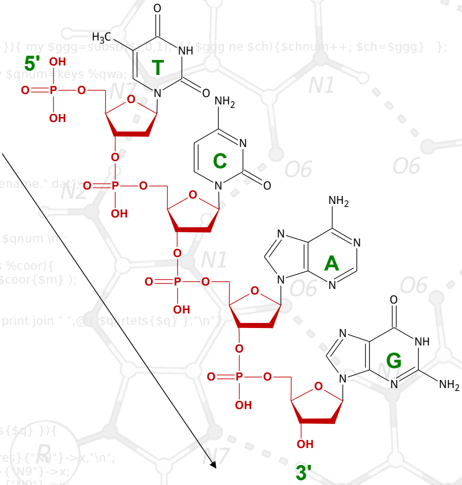
основание

Ортофосфорная кислота

гликозидная связь

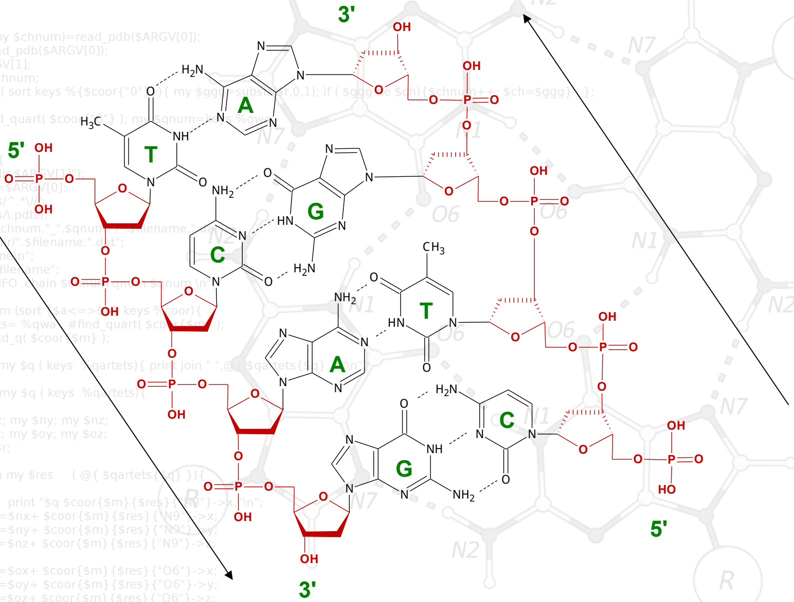
рибоза

Цепь ДНК

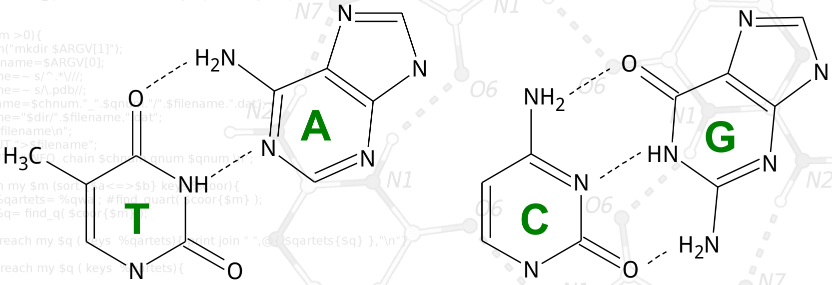


```
#!/usr/bin/perl
use Math::VectorReal qw( :all );

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}}){ my $ggg=substr($r,0,1); if ($ggg ne $ch){$chnum++; $ch=$ggg}   };

my %qwa=find_quart( $coor{"0"} ); my $qnum=keys %qwa;

if ($qnum >0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename=~ s/^.*\///;
$filename=~ s/\.pdb//;
#$filename=$chnum."_".$qnum."/".$filename.".dat";
$filename="$dir/".$filename.".dat";
print  "$filename\n";
open OUT,">$filename";
print OUT "#INFO  chain $chnum qnum $qnum \n";

  foreach my $m (sort {$a<=>$b} keys %coor){
    my  %qartets= %qwa ; #find_quart( $coor{$m} );
    my  %q= find_q( $coor{$m} );

#       foreach my $q ( keys  %qartets){ print join " ",@{ $qartets{$q} },"\n"};

        foreach my $q ( keys  %qartets){

         my $nx; my $ny; my $nz;
         my $ox; my $oy; my $oz;
            my $r;

         foreach my $res     ( @{ $qartets{$q} }){

#                    print "$q $coor{$m}{$res}{"N9"}->x,"\n";
            $nx=$nx+ $coor{$m}{$res}{"N9"}->x;
            $ny=$ny+ $coor{$m}{$res}{"N9"}->y;
            $nz=$nz+ $coor{$m}{$res}{"N9"}->z;

            $ox=$ox+ $coor{$m}{$res}{"O6"}->x;
            $oy=$oy+ $coor{$m}{$res}{"O6"}->y;
            $oz=$oz+ $coor{$m}{$res}{"O6"}->z;
```

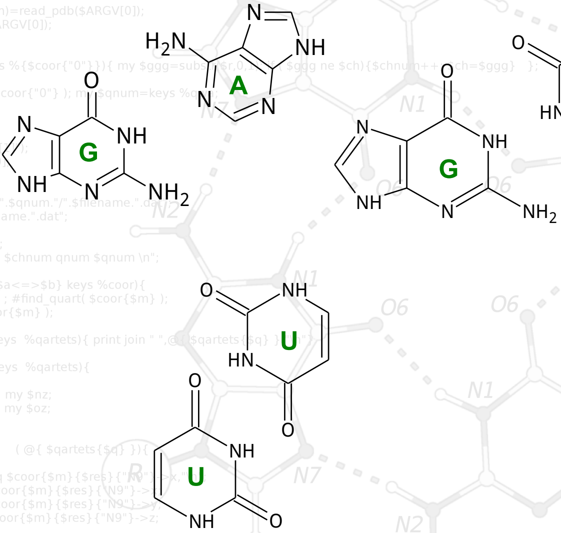
Двойная цепь ДНК



Канонические взаимодействия



Неканонические взаимодействия



Структура ДНК

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );


#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %

my %qwa=find_quart( $coor


if ($qnum >0){
#system("mkdir $ARGV[1]")
my $filename=$ARGV[0];
$filename=~ s/^.*\///;
$filename=~ s/\.pdb//;
#$filename=$chnum."_".$q
$filename="$dir/".$filename
print  "$filename\n";
open OUT,">$filename";
print OUT "#INFO  chain $ch

  foreach my $m (sort {$a<=
    my  %qartets= %qwa ; #f
    my  %q= find_q( $coor{$

#       foreach my $q ( keys

        foreach my $q ( keys

          my $nx; my $ny; my
          my $ox; my $oy; my
              my $r;

          foreach my $res     (

#                   print "$q $coor{$m}{$res}{"N9"}->x; \n";
            $nx=$nx+ $coor{$m}{$res}{"N9"}->x;
            $ny=$ny+ $coor{$m}{$res}{"N9"}->y;
            $nz=$nz+ $coor{$m}{$res}{"N9"}->z;

            $ox=$ox+ $coor{$m}{$res}{"O6"}->x;
            $oy=$oy+ $coor{$m}{$res}{"O6"}->y;
            $oz=$oz+ $coor{$m}{$res}{"O6"}->z;
```

Структура ДНК

Структура ДНК впервые была предложена Watson и Crick в 1953 на основе результатов РСА низкого разрешения

Основные свойства:

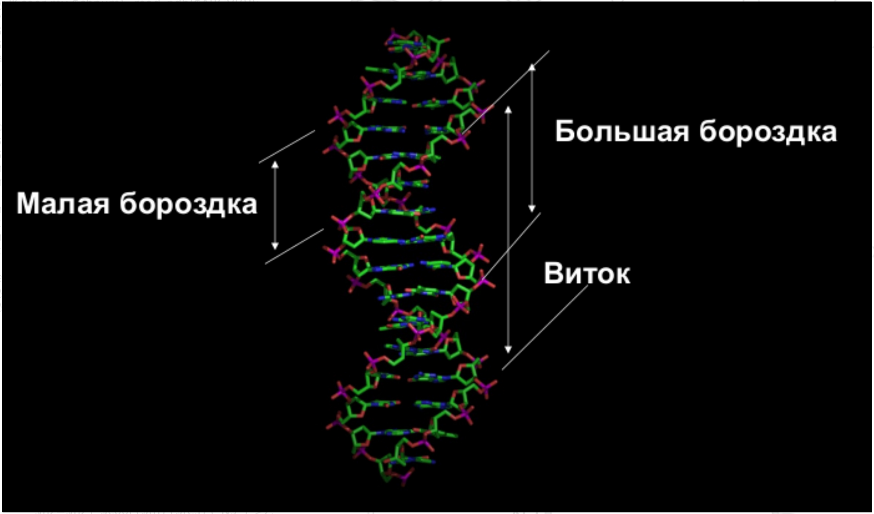
- Две антипараллельные цепи
- ДНК это двойная спираль
- Имеет две оси симметрии

Взаимодействия в ДНК

Два типа взаимодействий гетероциклических оснований в ДНК.

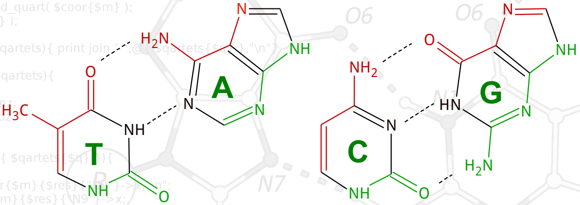
- Копланарные взаимодействия (в одной плоскости). В основном реализуются как водородные связи.
- Стопочные взаимодействия основаны на Ван-дер-Ваальсовых взаимодействиях.

Спираль ДНК

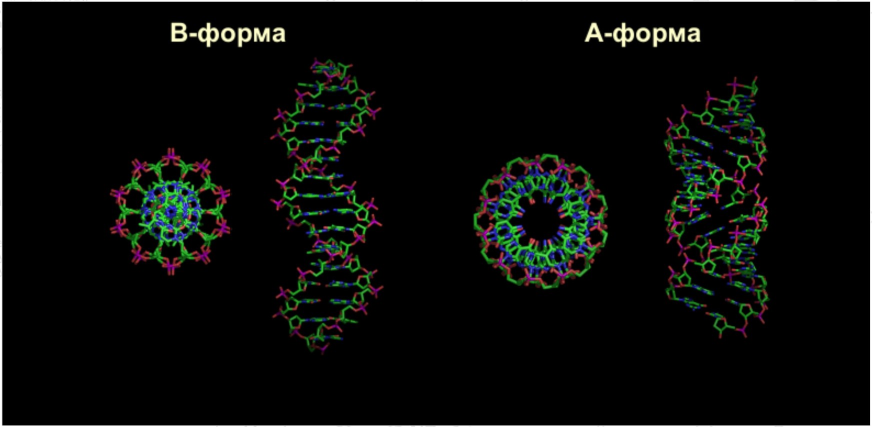


Атомы принадлежащие большой и малой бороздкам

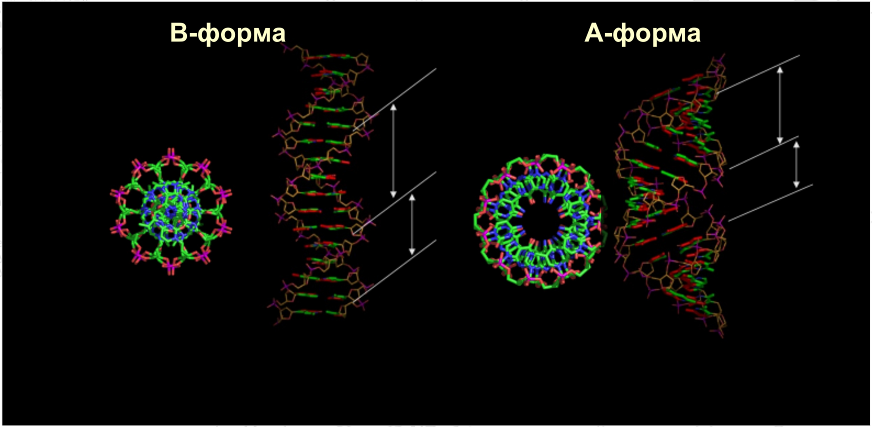
	Большая бороздка	**Малая бороздка**
Adenine	C6, N6, C5, N7, C5	C2, N3, C4, N9
Guanine	C6, O6, C5, N7, C8	C2, N2, N3, C4, N9
Cytosine	C6, C5, C4, N4	O2, N1, C2
Thymine	C6, C5, C4, O4, C5M	O2, C2, N1, C6



Регулярные формы спирали ДНК

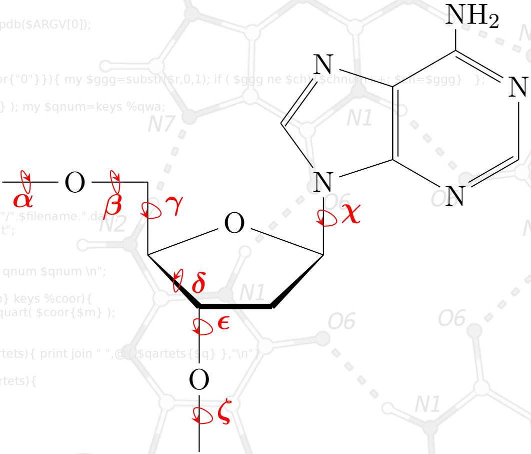


Регулярные формы спирали ДНК



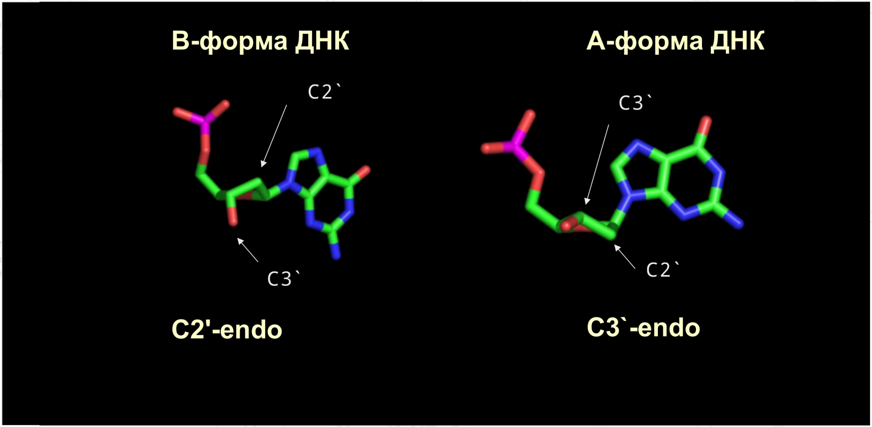
#!/usr/bin/perl
use Math::VectorReal qw(:all);
#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
# print "$q $coor{$m}{$res}{"N9"}->x,"\n";
$nx=$nx+ $coor{$m}{$res}{"N9"}->x;
$ny=$ny+ $coor{$m}{$res}{"N9"}->y;
$nz=$nz+ $coor{$m}{$res}{"N9"}->z;
$ox=$ox+ $coor{$m}{$res}{"O6"}->x;
$oy=$oy+ $coor{$m}{$res}{"O6"}->y;
$oz=$oz+ $coor{$m}{$res}{"O6"}->z;

Торсионные углы НК



Форма	α	β	γ	δ	ϵ	ζ	χ
А-ДНК	62	173	52	88/3	178	-50	-160
В-ДНК	63	171	54	123/131	155	-90	-117

Конформация дезоксирибозы



Условия существования различных форм

Разные формы ДНК переходят друг в друга при изменении условий внешней среды:

- B-форма стабильна при нормальных физиологических условиях
- Дегидратация, понижение относительной влажности до 0.75 инициирует переход B=>A

Пример: смеси вода-этанол(метанол) при росте доли спирта > 0.75 , переход B=>A

Переход ДНК из А в В форму в солевом растворе

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
[illegible]
[illegible]


#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %

my %qwa=find_quart( $coor


if ($qnum >0){
#system("mkdir $ARGV[1]")
my $filename=$ARGV[0];
$filename=~ s/^.*\///;
$filename=~ s/\.pdb//;
#$filename=$chnum."_".$q
$filename="$dir/".$filename
print  "$filename\n";
open OUT,">$filename";
print OUT "#INFO  chain $ch

  foreach my $m (sort {$a<=
    my  %qartets= %qwa ; #
    my  %q= find_q( $coor{$

#       foreach my $q ( keys

        foreach my $q ( keys

          my $nx; my $ny; my
          my $ox; my $oy; my
              my $r;

          foreach my $res     (

#                       print "$q $coor{$m}{$res}{"N9"}->x,\n";
             $nx=$nx+ $coor{$m}{$res}{"N9"}->x;
             $ny=$ny+ $coor{$m}{$res}{"N9"}->y;
             $nz=$nz+ $coor{$m}{$res}{"N9"}->z;

             $ox=$ox+ $coor{$m}{$res}{"O6"}->x;
             $oy=$oy+ $coor{$m}{$res}{"O6"}->y;
             $oz=$oz+ $coor{$m}{$res}{"O6"}->z;
```

Переход ДНК из A в B форму в солевом растворе

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );


#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %

my %qwa=find_quart( $coor


if ($qnum >0){
#system("mkdir $ARGV[1]")
my $filename=$ARGV[0];
$filename=~ s/^.*\///;
$filename=~ s/\.pdb//;
#$filename=$chnum."_".$q
$filename="$dir/".$filename
print  "$filename\n";
open OUT,">$filename";
print OUT "#INFO  chain $ch

  foreach my $m (sort {$a<=
    my  %qartets= %qwa ; #
    my  %q= find_q( $coor{$

#       foreach my $q ( keys

        foreach my $q ( keys

          my $nx; my $ny; my
          my $ox; my $oy; my
              my $r;

          foreach my $res     (

#                     print "$q $coor{$m}{$res}{"N9"}->x, \n";
             $nx=$nx+ $coor{$m}{$res}{"N9"}->x;
             $ny=$ny+ $coor{$m}{$res}{"N9"}->y;
             $nz=$nz+ $coor{$m}{$res}{"N9"}->z;

             $ox=$ox+ $coor{$m}{$res}{"O6"}->x;
             $oy=$oy+ $coor{$m}{$res}{"O6"}->y;
             $oz=$oz+ $coor{$m}{$res}{"O6"}->z;
```

Структура РНК

#!/usr/bin/perl
use Math::VectorReal qw(:all);
#(my %coor,my $
my %coor=read_p
my $dir=$ARGV[1
my $ch, my $chnu
foreach my $r (so
my %qwa=find_qu
if ($qnum >0){
#system("mkdir $
my $filename=$A
$filename=~ s/^.
$filename=~ s/\.p
#$filename=$chn
$filename="$dir/"
print "$filename\n
open OUT,">$filen
print OUT "#INFO
foreach my $m (s
my %qartets=
my %q= find_q
foreach my $
foreach my $
my $nx; m
my $ox; m
my $r;
foreach my
pri
$nx=$n
$ny=$n
$nz=$n
$ox=$ox+ $coor{$m}{$res}{"O6"}->x;
$oy=$oy+ $coor{$m}{$res}{"O6"}->y;
$oz=$oz+ $coor{$m}{$res}{"O6"}->z;

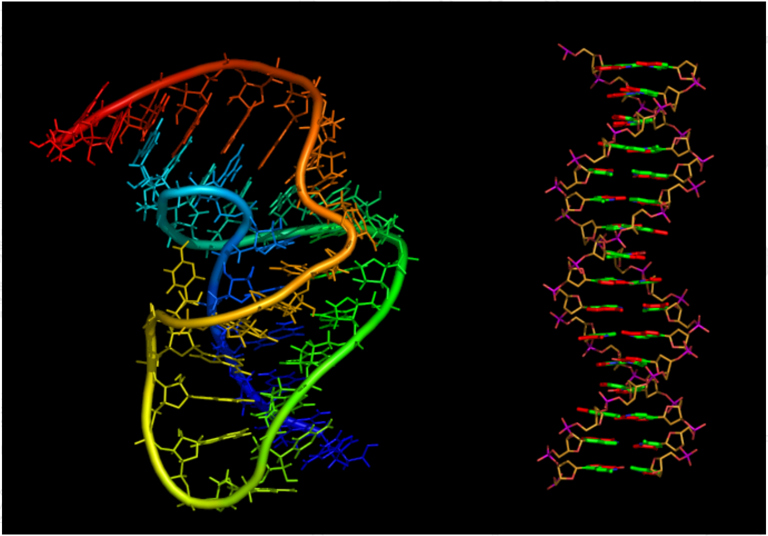
Структура РНК

Основные свойства.

- Одно-цепочечная молекула
- В клетке найдено множество видов РНК и каждый из них имеет специфичную функцию.

Основные типы: рРНК, мРНК, тРНК.

РНК и ДНК



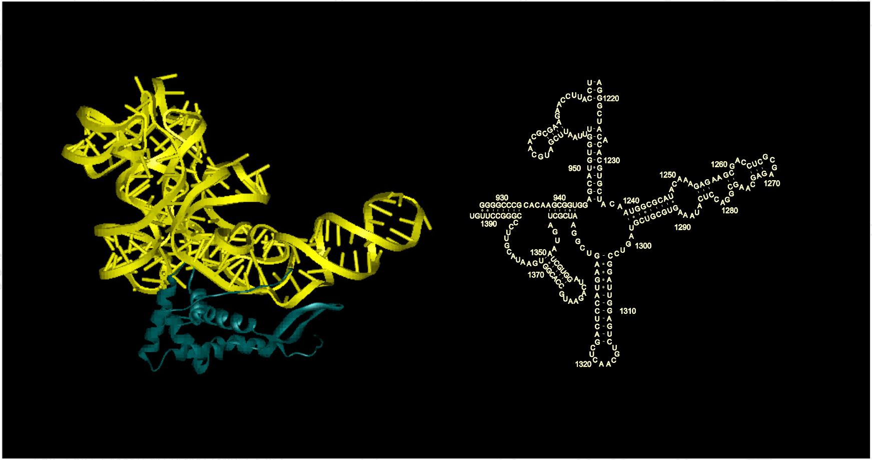
Почему важна структура РНК?

Структура РНК определяет её функцию:

- Регуляторная
- Структурная
- Каталитическая (рибозимы)

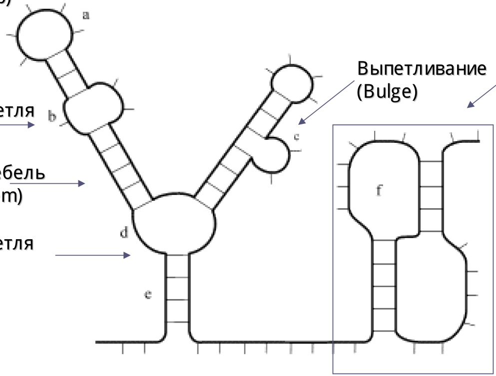
Некоторые вирусы имеют РНК геном (HIV,грипп).

Вторичная структура РНК



#!/usr/bin/perl
use Math::VectorReal qw(:all);
#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
$ny=$ny+ $coor{$m}{$res}{"N9"}->y;
$nz=$nz+ $coor{$m}{$res}{"N9"}->z;
$ox=$ox+ $coor{$m}{$res}{"O6"}->x;
$oy=$oy+ $coor{$m}{$res}{"O6"}->y;
$oz=$oz+ $coor{$m}{$res}{"O6"}->z;

Вторичная структура РНК



Петля (loop)

Внутренняя Петля (Internal loop)

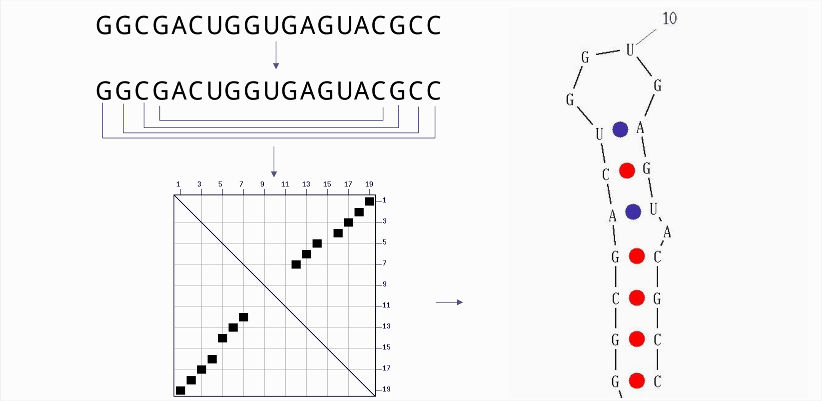
Стебель (stem)

мультипетля (junction)

Выпетливание (Bulge)

Псевдоузел (Pseudoknot)

Возможность предсказания вторичной структуры



GGCGACUGGUGAGUACGCC

GGCGACUGGUGAGUACGCC

Методы предсказания вторичной структуры РНК

- Поиск максимума количества пар оснований (алгоритм Nusinov)
- Минимизация энергии (алгоритм Зукера)
- Сравнительное предсказание на основе выравнивания последовательностей РНК (Viena RNA Package).

Расчёт энергии структуры по алгоритму Зукера

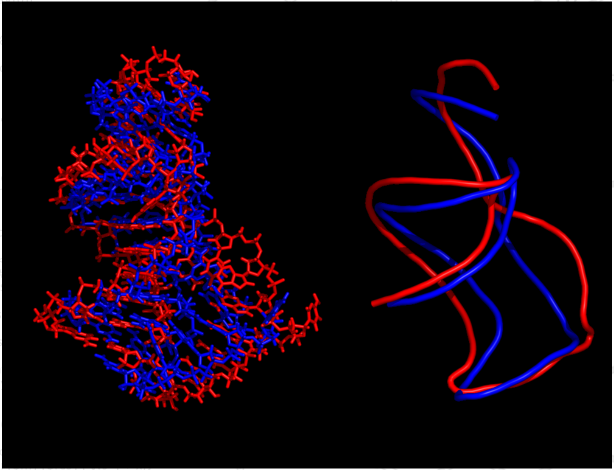
UU Петля +5.9 Ккал/моль
A A
G C Стэкинг+Пара –2*2.9 Ккал/моль
G C
A Выпетливание + 3.3 Ккал/моль
G C Стекинг+Пара –1.8 Ккал/моль
U A -0.9 Ккал/моль
A U -1.8 Ккал/моль
C G -2.1 Ккал/моль
A U 3′
A Неструктурированный 5`конец 0 Ккал/моль
A
5′

Δ G = -3.2 Ккал/моль

РНК, которые трудно предсказать алгоритмом Зукера

- РНК связанная с белками.
- Длинные РНК.
- Псевдоузлы.

Моделирование структуры



Вопросы?

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
use Math::Trig;
use strict;


#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}}){ my $ggg=substr($r,0,1); if ( $ggg ne $ch){$chnum++; $ch=$ggg}   };

my %qwa=find_quart( $coor{"0"} ); my $qnum=keys %qwa;


if ($qnum >0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename=~ s/^.*\///;
$filename=~ s/\.pdb//;
#$filename=$chnum."_".$qnum."/".$filename.".dat";
$filename="$dir/".$filename.".dat";
print  "$filename\n";
open OUT,">$filename";
print OUT "#INFO  chain $chnum qnum $qnum \n";

  foreach my $m (sort {$a<=>$b} keys %coor){
    my  %qartets= %qwa ; #find_quart( $coor{$m} );
    my  %q= find_q( $coor{$m} );

#       foreach my $q ( keys  %qartets){ print join " ",@{ $qartets{$q} },"\n"};

        foreach my $q ( keys  %qartets){

          my $nx; my $ny; my $nz;
          my $ox; my $oy; my $oz;
              my $r;

          foreach my $res     ( @{ $qartets{$q} }){

#                       print "$q $coor{$m}{$res}{"N9"}->x,"\n";
             $nx=$nx+ $coor{$m}{$res}{"N9"}->x;
             $ny=$ny+ $coor{$m}{$res}{"N9"}->y;
             $nz=$nz+ $coor{$m}{$res}{"N9"}->z;

             $ox=$ox+ $coor{$m}{$res}{"O6"}->x;
             $oy=$oy+ $coor{$m}{$res}{"O6"}->y;
             $oz=$oz+ $coor{$m}{$res}{"O6"}->z;
```