

# 天津医科大学实验课教案首页

(共 3 页、第 1 页)

课程名称：分子生物计算		实验名称：实验 9 制作酶切图谱	
教师姓名：伊现富	职称：讲师	教学日期：2019 年 10 月 30 日 13:30-15:10	
授课对象：生物医学工程与技术学院 2017 级生信班（本）		实验人数：28	
实验类型（验证型、综合型、设计型、创新型）：验证型		实验分组：一人一机	
学时数：2		教材版本：Perl 语言在生物信息学中的应用——基础篇	

## 实验目的与要求：

- 了解 REBASE 数据库中 bionet 格式。
- 熟悉模式匹配设计的特殊变量。
- 掌握正则表达式在酶切图谱制作中的应用。

## 实验内容及学时分配：

- (90') 实验操作：编写 Perl 程序制作酶切图谱。

## 主要仪器和实验材料：

- 主要仪器：一台安装有 Perl 语言（Linux 操作系统）的计算机。

## 实验重点、难点及解决策略：

- 重点难点：正则表达式的应用。
- 解决策略：通过演示进行学习，通过练习熟练掌握。

## 思考题：

- 举例说明正则表达式中的元字符
- 举例说明模式匹配中涉及的特殊变量。
- 解释逻辑操作符的求值顺序。

## 参考资料：

- Beginning Perl for Bioinformatics, James Tisdall, O'Reilly Media, 2001.
- Perl 语言入门（第六版），Randal L. Schwartz, brian d foy & Tom Phoenix 著，盛春 译，东南大学出版社，2012。
- Mastering Perl for Bioinformatics, James Tisdall, O'Reilly Media, 2003.
- 维基百科等网络资源。

主任签字：

年 月 日

教务处制

## 一、实验操作 (90 分钟)

### 1. 把 IUB 代码翻译成正则表达式

```
sub IUB_to_regexp {
    my ($iub) = @_;
    my $regular_expression = '';
    my %iub2character_class = (
        A => 'A',
        C => 'C',
        G => 'G',
        T => 'T',
        R => '[GA]',
        Y => '[CT]',
        M => '[AC]',
        K => '[GT]',
        S => '[GC]',
        W => '[AT]',
        B => '[CGT]',
        D => '[AGT]',
        H => '[ACT]',
        V => '[ACG]',
        N => '[ACGT]',
    );
    $iub =~ s/\^//g;
    for ( my $i = 0 ; $i < length($iub) ; ++$i ) {
        $regular_expression .= $iub2character_class{ substr( $iub, $i, 1 ) };
    }
    return $regular_expression;
}
```

### 2. 解析 REBASE 中的 bionet 数据文件

```
sub parseREBASE {
    my ($rebasefile) = @_;
    use strict; use warnings; use BeginPerlBioinfo;
    my @rebasefile = ();
    my %rebase_hash = ();
    my $name; my $site; my $regexp;
    my $rebase_filehandle = open_file($rebasefile);
    while (<$rebase_filehandle>) {
        ( 1 .. /Rich Roberts/ ) and next;
        /\s*$/ and next;
        my @fields = split( " ", $_ );
        $name = shift @fields;
        $site = pop @fields;
        $regexp = IUB_to_regexp($site);
        $rebase_hash{$name} = "$site $regexp";
    }
    return %rebase_hash;
}
```

## 3. 根据用户的查询制作酶切图谱

```
#!/usr/bin/perl
use strict; use warnings;
use BeginPerlBioinfo;
my %rebase_hash      = ();
my @file_data        = ();
my $query            = '';
my $dna               = '';
my $recognition_site = '';
my $regex             = '';
my @locations         = ();
@file_data = get_file_data("sample.dna");
$dna = extract_sequence_from_fasta_data(@file_data);
%rebase_hash = parseREBASE('bionet');
do {
    print "Search for what restriction site for (or quit)?: ";
    $query = <STDIN>;
    chomp $query;
    if ( $query =~ /^\\s*$/ ) {
        exit;
    }
    if ( exists $rebase_hash{$query} ) {
        ( $recognition_site, $regex ) = split( " ", $rebase_hash{$query} );
        @locations = match_positions( $regex, $dna );
        if (@locations) {
            print "Searching for $query $recognition_site $regex\\n";
            print "A restriction site for $query at locations:\\n";
            print join( " ", @locations ), "\\n";
        }
        else {
            print "A restriction enzyme $query is not in the DNA:
\\n";
        }
    }
    print "\\n";
} until ( $query =~ /quit/ );
exit;
sub match_positions {
    my ( $regex, $sequence ) = @_;
    use strict;
    use BeginPerlBioinfo;    # see Chapter 6 about this module
    my @positions = ();
    while ( $sequence =~ /$regex/ig ) {
        push( @positions, pos($sequence) - length($&) + 1 );
    }
    return @positions;
}
```