

天津医科大学理论课教案首页

(共3页、第1页)

课程名称：分子生物计算

课程内容/章节：突变和随机化 / 第7章

教师姓名：伊现富

职称：讲师

教学日期：2016年12月7&14日 15:30-17:30

授课对象：生物医学工程与技术学院 2013 级生信班（本）

听课人数：30

授课方式：理论讲授

学时数：6

教材版本：Perl 语言在生物信息学中的应用——基础篇

教学目的与要求（分掌握、熟悉、了解、自学四个层次）：

- 掌握：随机选取数组元素的方法；随机选取字符串位置的方法；随机选取两个整数间数字的方法。
- 熟悉：随机化在模拟 DNA 突变和生成随机 DNA 序列中的应用。
- 了解：随机数生成器及其种子。
- 自学：自上而下和自下而上的程序设计理念。

授课内容及学时分配：

- (15') 引言与导入：回顾突变的概念、原因和类型，介绍随机和模拟及其在 DNA 突变中的应用。
- (10') 随机数生成器：介绍随机数生成器、伪随机数以及种子的相关概念。
- (75') 随机化程序：通过生成随机句子的 Perl 程序讲解随机化在 Perl 语言中的实现。
- (50') 模拟 DNA 突变：通过模拟 DNA 突变的 Perl 程序讲解随机化在模拟 DNA 突变中的应用。
- (50') 生成随机 DNA：通过生成随机 DNA 序列的 Perl 程序讲解随机化在生成随机 DNA 序列中的应用。
- (90') 分析 DNA：详细讲解如何基于已有的知识生成随机 DNA 序列并对它们进行相似性比较。
- (10') 总结与答疑：总结授课内容中的知识点与技能，解答学生疑问。

教学重点、难点及解决策略：

- 重点：随机选取数组元素；随机选取字符串位置；随机选取两个整数间的数字。
- 难点：随机选取数组元素；嵌套循环的工作步骤。
- 解决策略：通过实例演示帮助学生理解、记忆。

专业外语词汇或术语：

突变 (mutation)

随机数生成器 (random number generator)

随机性 (randomness)

伪随机性 (pseudorandomness)

模拟 (simulation)

种子 (seed)

辅助教学情况：

- 多媒体：突变的类型；随机化在 Perl 语言中的实现。
- 板书：模拟 DNA 突变的步骤；嵌套循环的工作步骤。
- 演示：通过随机化模拟 DNA 突变、生成随机 DNA 序列。

复习思考题：

- 如何设置 Perl 语言随机数生成器的种子？
- 如何随机选取数组的元素？
- 如何随机选取字符串中的位置？
- 如何随机选取两个整数间的一个数字？
- 比较自上而下和自下而上的程序设计理念。
- 解释嵌套循环的工作步骤。

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、引言与导入 (15 分钟)

1. 突变

(1) 简介

- 定义：突变，即基因突变，指细胞中的遗传基因发生的改变
- 原因：复制错误、化学毒性、物理辐射、生物病毒……
- 影响：细胞死亡或者癌变；进化的“推动力”（双刃剑！）
- 类型：点突变、插入、删除、重复、倒位、易位……

	No mutation	Point mutations			
		Silent	Nonsense	Missense	
				conservative	non-conservative
DNA level	TTC	TTT	ATC	TCC	TGC
mRNA level	AAG	AAA	UAG	AGG	ACG
protein level	Lys	Lys	STOP	Arg	Thr
					
				basic	polar

(2) 点突变

- 简介：单一核苷酸替换成了另一核苷酸（狭义 vs. 广义）
- 分类
 - 转换（嘌呤变嘌呤，嘧啶变嘧啶），颠换（嘌呤变嘧啶，嘧啶变嘌呤）
 - 同义突变（氨基酸产物不变），错义突变（氨基酸产物改变），无义突变（氨基酸变成终止密码）
- 随机：一个不定因子不断产生的重复过程，可能遵循某个概率分布（vs. 任意）；（进化论）随机突变
- 模拟：通过随机化模拟 DNA 突变
- 即将学习
 - 随机选取数组元素，随机选取字符串位置
 - 模拟 DNA 突变：随机选取 DNA 中的一个核苷酸突变成一个随机的核苷酸
 - 根据要求生成随机 DNA 序列数据集
 - 重复突变 DNA 来研究突变随时间累积的影响

二、随机数生成器 (10 分钟)

- 随机数生成器：通过一些算法、物理讯号、环境噪音等来产生看起来似乎没有关联性的数列的方法或装置（丢硬币、掷骰子、洗牌）
- 伪随机数：重复周期比较大的数列，并不是真正的随机数，按一定的算法和种子值生成
- 种子：种子改变，伪随机数随之改变；种子本身应该是随机选择的

三、随机化程序 (75 分钟)

- Perl 程序 7.1：通过随机选取名词、动词和介词构造句子
- 设置随机数种子：`srand(time | $$);`
 - `time`：时间；`$$`：PID；`|`：位或
 - `srand`；会自动设置种子；`rand` 会自动调用 `srand` 设置种子
- 【重点、难点】随机选取数组元素（比较并理解三种不同的写法）
 - `$verbs[int(rand(scalar @verbs))]`（由内而外层层解析）
 - `$verbs[int rand scalar @verbs]`
 - `$verbs[rand @verbs]`
- 【重点、难点】`$verbs[rand @verbs]`
 - `rand` 期望一个标量值，所以会把 `@verbs` 放在标量上下文中进行求值，返回数组元素个数
 - 数组的下标总是整数值，所以当它需要下标时，会自动提取小数的整数部分，因此不再需要 `int`

四、模拟 DNA 突变 (50 分钟)

1. **【重点】** 编写子程序 (对每一个子程序都单独进行测试)
 - 随机选取 DNA 序列中的一个位置 (借鉴随机选取数组元素的策略)
 - 随机选取一个核苷酸 (每一个子程序都可能有改进的空间)
 - 实现 DNA 序列上的一次突变 (充分利用已经编写完成的子程序)
2. 组合子程序 (体会子程序的便利; 总结分割子程序的经验)
 - Perl 程序 7.2: 组合需要的子程序模拟 DNA 突变
 - 改进程序: 保证突变成和原核苷酸不同的核苷酸
3. 声明变量
 - 在程序顶部统一声明 vs. 第一次使用时声明
 - 局部变量 (在代码块中声明) vs. 全局变量 (在主程序中声明)

五、生成随机 DNA (50 分钟)

1. 设计理念 (理念不同, 目的相同; 比较并练习不同的设计理念)
 - 自上而下: 从主程序开始, 当需要子程序再进行编写
 - 自下而上: 从子程序开始, 逐步组装出完整的主程序
2. 伪代码
 - 总体目标: 生成一系列长短不一的随机 DNA 片段
 - 自上而下 (将大任务逐步分割成容易实现的小目标)
 - (1) 生成一系列长短不一的随机 DNA 片段
 - (2) 编写 make_random_DNA_set 子程序
 - (3) 编写 make_random_DNA 子程序
 - (4) 编写 randomnucleotide 子程序
3. Perl 程序 7.3: 根据要求生成一系列长短不一的随机 DNA 片段
 - **【重点】** 随机选取两个整数间的一个整数 (妙用 rand 函数; 理解 +1 的原因)


```
- int( rand( $maxlength - $minlength + 1 ) ) + $minlength
```

六、分析 DNA (90 分钟)

1. 问题: 两条 DNA 序列的相似性如何?
2. Perl 程序 7.4: 计算一个 DNA 数据集中两两之间相似性百分比的平均值
3. **【难点】** 嵌套循环 (注意内外两层循环初始值和终止值的差别)

七、总结与答疑 (10 分钟)

1. 知识点
 - 随机: 数组元素, 字符串位置
 - 随机数生成器: 伪随机, 种子
 - 程序设计理念: 自上而下, 自下而上
 - Perl 语言: 变量声明, 嵌套循环
2. 技能
 - 熟练使用 Perl 语言中的随机数生成器
 - 熟悉自上而下和自下而上的设计理念
 - 用 Perl 编写 DNA 突变相关的程序

