

第一章：计算机基础知识

姓名：张治卓 学号：517111910078

1. GNU是什么意思？GNU Linux与我们安装的Linux有什么区别？你都知道哪些操作系统和哪些编程语言？请分别列出3-4种。
答: GNU操作系统是一种由自由软件构成的类 Unix 操作系统，该系统基于 Linux 内核，目标在于建立一个完全相容于UNIX的自由软件环境。
操作系统：Windows，Linux，BSD，Mac OS X
编程语言：C/C++，Python，Basic，R
2. 我们都知道计算机中所有的对象都是用二进制0/1表示的，如果要表示A/C/G/T四种碱基，最少该用几位的二进制表示？如何表示？20种氨基酸呢？
答：（1）2位，分别可用00,10,01,11
（2）5位，共有32种组合取其中20种即可。
3. 对于一个已经存在的10进制整数，我们可以通过反复对10取余运算（`mod`）可得到该整数的每一位数值。同样的，其二进制、八进制、十六进制也可以通过这样的取余运算得到。当然，像我们课上介绍的一样，二进制转成八进制和十六进制是非常简单的。这里给出几个十进制的数值，写出其二进制、八进制和十六进制的表示。

- 235

◦ 62
- 223

答：

十进制	二进制	八进制	十六进制
235	11101011	353	EB
62	00111110	76	3E
223	11011111	337	DF

4. 对一个FASTQ文件，其每一位的质量分值是用ASCII码表示的。对于最常用的Illumina的Phred+33体系，其计算过程是：

$$Q = 33 + (-10) \times \log_{10} P$$

这里的 P 是测序错误的概率，然后再将 Q 转换为对应的ASCII码字符。那么，对于下面的FASTQ文件中的测序序列（其中第二行为序列，第四行为序列质量打分对应的ASCII字符，`@`和`+`开始的行为序列名称等信息），请计算完全没有测序错误的概率。

```
@M05025
CTCAG
+
C:)?7
```

答：先将`C:)?7`对应的ASCII码转换得到对应的数字为`67、58、41、63、55`，然后代入公式 $Q = 33 + (-10) \times \log_{10} P$ 得到`CTCAG`对应的测序错误概率分别为 4.00×10^{-4} ， 3.16×10^{-3} ， 1.58×10^{-1} ， 1.00×10^{-3} ， 6.31×10^{-3} ，因此完全没有测序错误的概率为 $(1 - 4.00 \times 10^{-4}) \times (1 - 3.16 \times 10^{-3}) \times (1 - 1.58 \times 10^{-1}) \times (1 - 1.00 \times 10^{-3}) \times (1 - 6.31 \times 10^{-3}) = 0.833$

5. 我们知道，大部分的浮点数在计算机中是不能精确表示的。请问为什么？并举例说明。
答：因为计算机使用二进制数来记录数字，而很多浮点数的小数位无法被有限二进制位数表达，例如 $0.1 = 1 \div (\frac{1}{2})^4 + 1 \div (\frac{1}{2})^5 + 1 \div (\frac{1}{2})^8 + \dots$ 。因此很多浮点数都无法被精确表示。
6. 阅读博文[浮点数是如何存储的](#)，然后写出下面的浮点数在计算机中的IEEE标准二进制存储方法：

◦ 13.1

◦ 0.65

答：

13.1

S=0，E=10000010，M=10100011001100110011001

所以为01000001010100011001100110011001

0.65

S=0，E=01111110，M=01001100110011001100110

所以为001111111001001100110011001100110
7. 一个8位二进制补码表示的最大正整数是什么？最小的负整数又是什么？请分别写出其二进制和十进制的数值。
答：最大正整数为+127，最小负整数为-128

+127	-128
01111111	10000000

8. 将下列二进制补码转换为8-bit的二进制补码。

◦ 1010

◦ 011001

◦ 1111111000

◦ 01

答：
- | 原二进制补码 | 8-bit补码 |
|--------|----------|
| 1010 | 11111010 |

原二进制补码	8-bit补码
011001	00011001
1111111000	11111000
01	00000001

9. 阅读博文[彻底了解IP地址](#)后，计算IP地址为192.168.5.100，子网掩码为255.255.128.0的网络地址、广播地址和网段。
答：

IP地址	192.168.5.100	11000000•10101000•00000101•01100100
子网掩码	255.255.128.0	11111111•11111111•10000000•00000000
网络地址	192.168.0.0	11000000•10101000•00000000•00000000
广播地址	192.168.0.255	11000000•10101000•00000000•11111111

网段：192.168.0.1-192.168.0.254

10. 请在你的计算机上安装一个Linux的发行版本，如Ubuntu、CentOS或者Fedora，并记录详细的安装流程。命令 `lsb_release -a` 可以输出你安装的发行版本的信息，请查看并在这里输出结果。你可以用（1）虚拟机安装，可用（2）用容器docker安装，也可以（3）安装双系统。
答：

```
Distributor ID:    Ubuntu
Description:      Ubuntu 18.04.2 LTS
Release:          18.04
Codename:         bionic
```

第一章附录1：补码表示法

对于一个有符号整数（signed integer），可以用原码、反码（one’ s complement）和补码（two’ s complement）表示（为了方便阐述，我们这里采用8bits表示方法）：

- 1. 第一位为符号位，0为正数，1为负数；
- 2. 对于正整数来说，其原码、反码和补码是相同的
- 3. 对于负整数来说，原码是将其相反数原码的符号位取1,其他位不变；其反码则是将其相反数的所有位取反；补码则是在反码的基础上加1。

举例

- 对于正整数28，其原码是00011100，其反码和补码也都是00011100；
- 对于负整数-28,其原码是将+28的原码的符号位取1,也就是10011100，其反码是将+28原码所有位取反，也就是11100011，其补码则是将其反码加1,则为11100100；
- 对于0来说，不存在+0和-0的差别，原码、反码和补码都表示为00000000；
- -128的补码表示为10000000；
- 因此8bits的整数取值范围是[－128, 127]。