**初级**

1. 写程序将” Hello World”打印到屏幕。

2. 写程序输入用户的姓名并用该姓名和他打招呼。

3. 修改上一个程序，使得仅可以与Alice和Bob这两个用户用其姓名与之打招呼。

4. 写程序输入一个数n并打印出从1到n的和。

5. 修改上个程序，使得求和的数只包含3或5的倍数，例如n=17，则求和的数为：3, 5, 6, 9, 10, 12, 15。

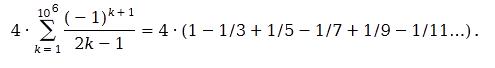
6. 写个程序，要求用户输入一个数n，并概率性的选择是计算1到n的和还是计算1到n的乘积。

7. 写程序打印出12×12乘法表。

8. 写程序打印所有的素数。（注意：如果你用的编程语言不支持任意大小的数，那么打印出所有你能表示的素数，包括最大数）

9. 写一个竞猜游戏，用户必须猜一个秘密的数字，在每次猜完后程序会告诉用户他猜的数是太大了还是太小了，直到猜测正确，最后打印出猜测的次数。如果用户连续猜测同一个数字则只算一次。

10. 写个程序打印出接下来的20个闰年。

11. 写程序计算：  
[](http://jbcdn2.b0.upaiyun.com/2013/12/20131209145605.jpg)

**列表list和字符串**

1. 写一个函数，返回列表中最大的数。

2. 写函数逆转列表，最好是原地逆转。

3. 写个函数检查指定的元素是否出现在列表中。

4. 写个函数返回列表中奇数位置的所有元素。

5. 写个函数计算列表的运行花费总和（the running total）。

6. 写个函数测试一个字符串是否是回文。

7. 写三个函数来计算列表中数字的和：分别用for循环，while循环和递归完成。

8. 写个函数on\_all遍历列表中的每个元素，打印出开始的20个完全平方数。

9. 写个函数连接两个列表。

10. 写个函数交替合并两个列表，例如：[a,b,c], [1,2,3] → [a,1,b,2,c,3]。

11. 写个函数合并两个有序的列表。

12. 写个函数计算前100个Fibonacci数的列表。

13. 写个函数，返回指定数的各位数字的列表。

14. 写个函数对两个数进行加减乘，使用各个位上的数字表示的列表实现并返回一个新的数字列表，如果你有信心可以实现Karatsuba乘法。尝试[不同的基数](https://en.wikipedia.org/wiki/Radix)，如果你关心速度可以比较下哪个是最佳基数。

15. 实现下面的排序算法：选择排序，插入排序，归并排序，快速排序，[臭皮匠排序](http://zh.wikipedia.org/wiki/%E8%87%AD%E7%9A%AE%E5%8C%A0%E6%8E%92%E5%BA%8F)（Stooge Sort）。具体的描述见Wikipedia。

16. 实现二分查找。

17. 写个函数，给定一个字符串列表并按下面表示打印出来，一行一个打印在矩形框中。例如列表["Hello", "World", "in", "a", "frame"] 打印的结果是：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | \*\*\*\*\*\*\*\*\*  \* Hello \*  \* World \*  \* in    \*  \* a     \*  \* frame \*  \*\*\*\*\*\*\*\*\* |

18. 写函数将一段文本text翻译为Pig Latin返回，英语翻译为Pig Latin的规则是：取出每个单词的首个字母，追加’ay’后再放到该单词的末尾。例如“The quick brown fox” 翻译后就变成了 “Hetay uickqay rownbay oxfay”。

**中级**

1. 写程序在1,2,…,9（保持这个顺序）之间可任意放+或-或都不放使其结果等于100，输出所有可能的放法。例如：1 + 2 + 3 – 4 + 5 + 6 + 78 + 9 = 100。

2. 写程序以一个假想行星的一年持续时间作为输入，产生一个闰年规则，最大限度的减少与该行星的太阳年的差异。

3. 实现数据结构图，允许修改（插入，删除），并能够存储边和节点的值。可能使用（node, edgelist）字典对表示完成该功能最容易。

4，写个函数生成图的DOT表示（译者注：[DOT](https://zh.wikipedia.org/zh/DOT%E8%AF%AD%E8%A8%80)语言是一种文本图形描述语言，它提供了一种简单的描述图形的方法，并且可以为人类和计算机程序所理解。）。

5. 写程序自动的给你生成文章：

（1）使用一个样例文本sample创建有向（多）图，其中文本单词作为节点，如果文本中u后面是v则u和v之间有一条有向边，多次出现生成多条边。

（2）在该图中做随机遍历：从一个随机的节点开始选择一个随机的后继节点，如果没有后继节点就随机的选择另一个节点。

6. 写程序自动的将英文文本转换为摩尔斯电码（译者注： 摩斯码（[Morse Code](http://zh.wikipedia.org/wiki/%E6%91%A9%E5%B0%94%E6%96%AF%E7%94%B5%E7%A0%81)）是一种时通时断的信号代码，通过不同的排列顺序来表达不同的英文字母、数字和标点符号）或者相反。

7. 写程序找出给定字符串的最长回文子串，尽可能高效的实现。

**高级**

1. 给定两个字符串，写程序高效的找出最长的公共子串。

2. 给定一个整数数组，写程序高效的查询：比位置i处的数稍大的最近邻数，这里的距离是指数组下标的绝对差。例如数组[1,4,3,2,5,7]，比4大的最近的数是5。先用线性时间做预处理，然后用常数时间做查询。

3. 给定两个字符串，使用字符插入和删除将其中的一个字符串转变成另一个，输出最短的插入和删除序列。

4. 写个函数实现两个矩阵相乘。尽可能高效的实现并使用较好的线性代数库（linear algebra library，具体可看[这里](http://en.wikipedia.org/wiki/Comparison_of_linear_algebra_libraries)）进行性能比较。你也许想读一下[Strassen’s algorithm](https://en.wikipedia.org/wiki/Strassen_algorithm)和CPU缓存的影响，尝试不同的矩阵布局，看看发生了什么。

5. 给定d维矩阵框集合，写程序计算它们交集的体积，从2维开始逐步计算。