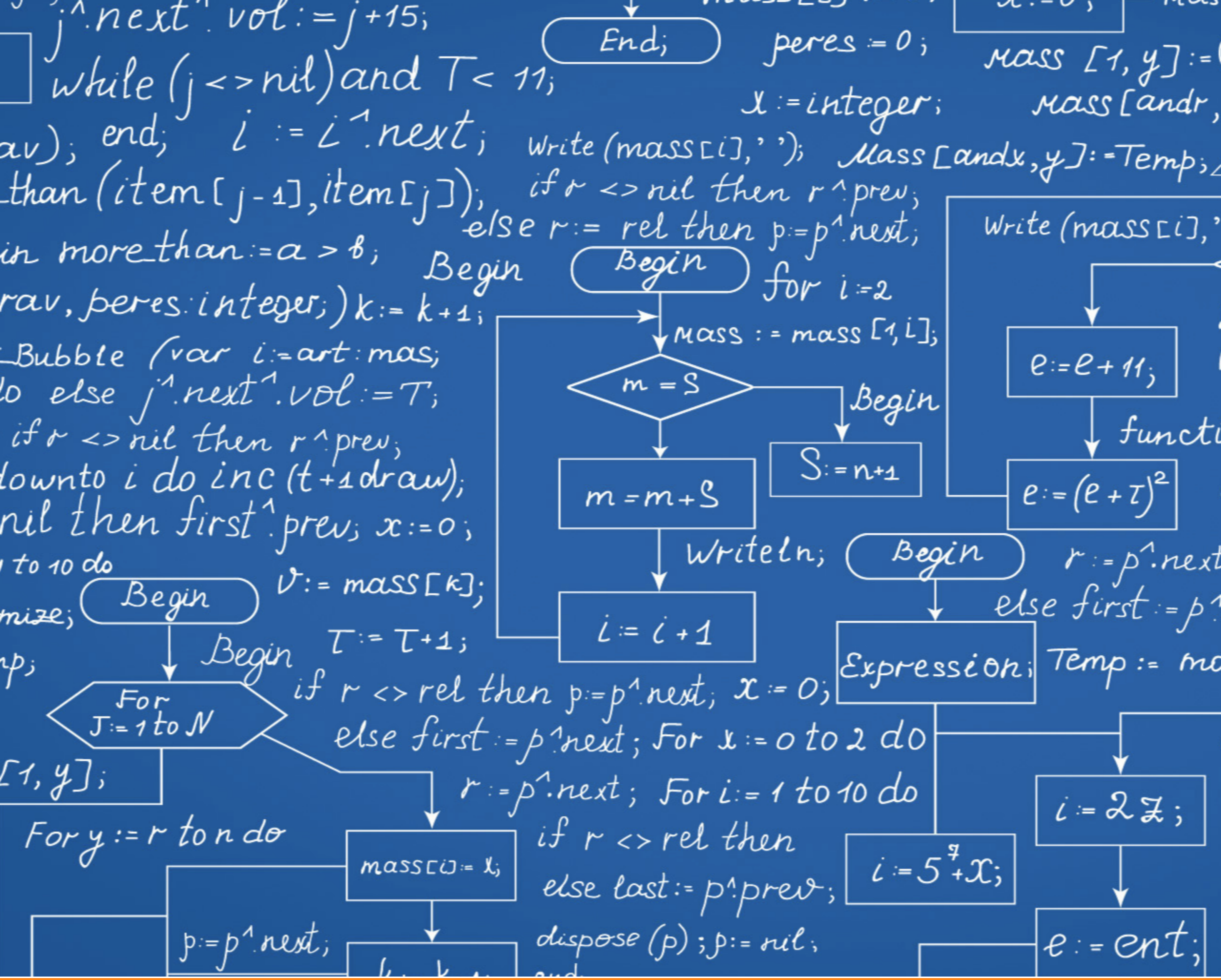

何老师算法课笔记

还没有想好的副标题

作者：计卓 1801 全体

组织：华中科技大学

时间：2020 年 11 月 9 日

版本：1.0.0

目录

第 1 章 分治法之大数乘法

内容提要

- 问题背景
- 算法 1
- 算法 2
- 伪代码及复杂度分析

1.1 问题描述

给定两个大数 A 和 B, 试计算 $A * B$. 其中 A 和 B 分别表示为 $A = a_n a_{n-1} a_{n-2} ... a_2 a_1$, $B = b_n b_{n-1} b_{n-2} ... b_2 b_1$.

引理 1.1. "$A + B$"的复杂度

计算 $A + B$，其复杂度为 $O(n)$, 其中 n 为 A 和 B 的十进制位数。

定理 1.1. "$A * B$"的朴素算法复杂度

朴素算法下，将 A 与 B 的各位相乘，再相加各次相乘的结果。不难得出，这一过程需要进行 n 次基本乘法与 $n + 1$ 次加法。根据引理1.1，朴素算法下计算 $A * B$ 的时间复杂度为 $O(n^2)$.

由定理1.1和引理1.1可知，朴素的大数乘法相比加法运算在时间复杂度上高出了一个量级。由于乘法在计算机中大量存在，我们希望找到更好的算法来降低乘法计算的时间复杂度，以提升计算机的性能。分治法为我们提供了一条途径。

1.2 直接分治法

1.2.1 算法描述

这是一种简单的分治方法，将两个大数分为前后两部分，进行相乘。不失一般性，这里假设 n 为偶数。将 A 与 B 分割为 A_2,A_1,B_2,B_1, 即：

$$A_2 = a_n a_{n-1} ... a_{\frac{n}{2}+2} a_{\frac{n}{2}+1}$$

$$A_1 = a_{\frac{n}{2}} a_{\frac{n}{2}-1} ... a_2 a_1$$

则 A 被表示为 $A = A_2 * 2^{\frac{n}{2}} + A_1$. 同理，$B$ 可以进行类似的分割，即：

$$B_2 = b_n b_{n-1} ... b_{\frac{n}{2}+2} b_{\frac{n}{2}+1}$$

$$B_1 = b_{\frac{n}{2}} b_{\frac{n}{2}-1} ... b_2 b_1$$

B 被表示为 $B = B_2 * 2^{\frac{n}{2}} + B_1$.

对我们的问题而言，计算 $A*B$ 则可以表示为：

$$\begin{aligned} A*B &= (A_2 * 2^{\frac{n}{2}} + A_1) * (b_{\frac{n}{2}} b_{\frac{n}{2}-1} ... b_2 b_1) \\ &= A_2 B_2 * 2^n + (A_2 B_1 + A_1 B_2) * 2^{\frac{n}{2}} + A_1 B_1 \end{aligned}$$

Data: this text
Result: how to write algorithm with LaTeX2e
initialization;
while *not at end of this document* **do**
  read current;
  if *understand* **then**
    go to next section;
    current section becomes this one;
  else
    go back to the beginning of current section;
  end
end

Algorithm 1: How to write algorithms

参考文献