哈尔滨工业大学计算机科学与技术学院

实验报告

课程名称：生物信息学

课程类型：生物信息学专业课

实验项目名称：

大规模基因组序列表示与索引系统设计与实验

实验题目：CSA-BWT

班级：1803401

学号：1161140116

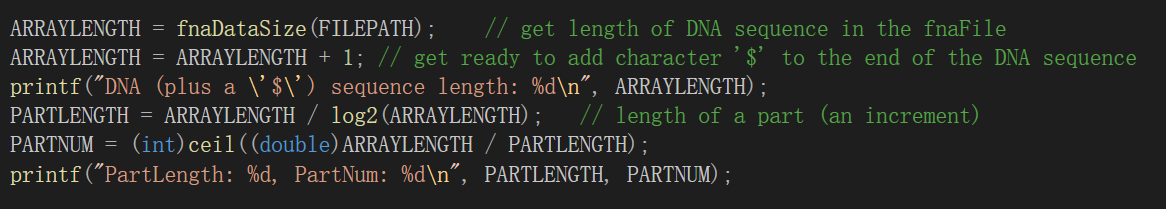
姓名：陈少杰

|  |  |  |
| --- | --- | --- |
| 设计成绩 | 报告成绩 | 指导老师 |
|  |  |  |

1. 实验目的

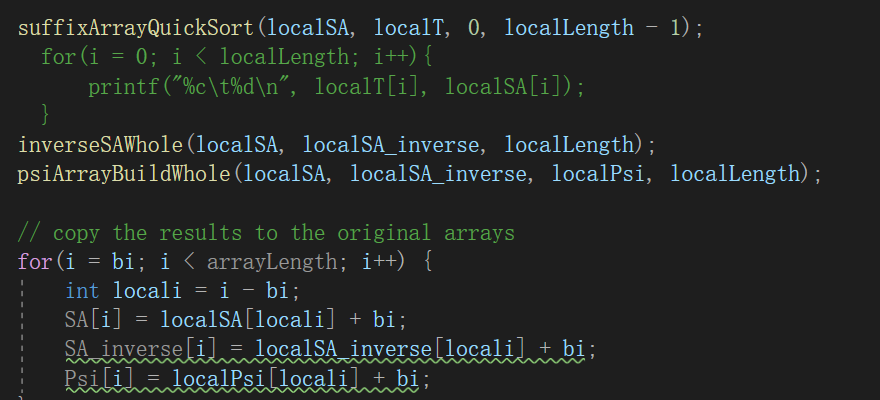
1.给定基因组序列（大肠杆菌E. Coli, 长度~5Mbp） 作为输入

2.建立算法和系统，可以为基因组序列建立自索 引表示

1. 实验原理
2. 把序列分为从T1到Tn的n个片段，每个片段长度l为n/logn，

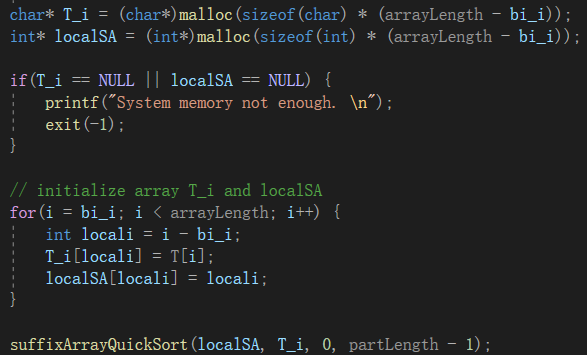
计算l

1. 计算最后一个片段Tn/l的ψ值，这一步为base step



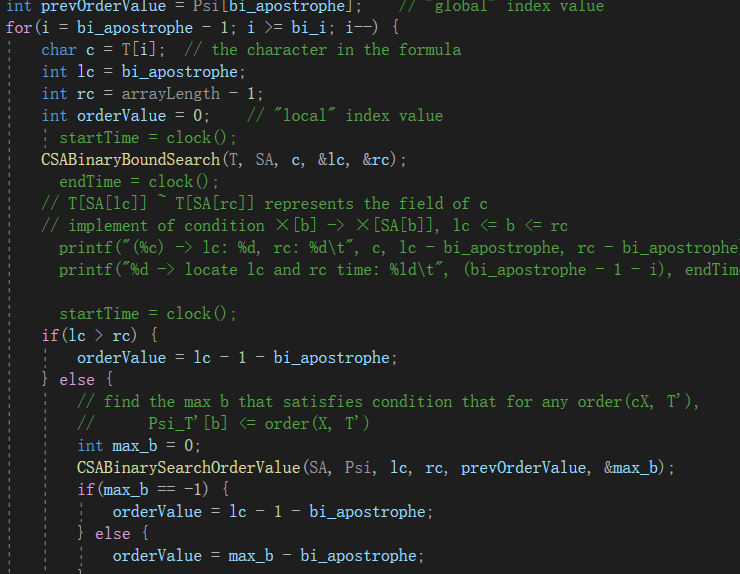
用快排计算SA，再通过ψ[i]=SA^-1[SA[i]+1]得到第n段的ψ序列，O（llogl）

1. 合并步骤，构建好ψ的序列为T’，下一片段为Ti，通过增量构建来计算TiT’的ψ值，再循环3直到完成全部序列的CSA构建
2. 对于suf1,…,sufl排序，找到各自顺序



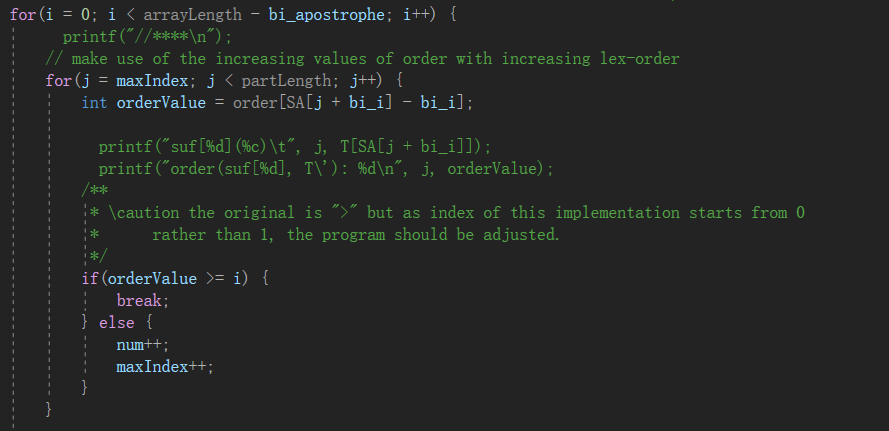
对这l个suf快排，O(llogl)

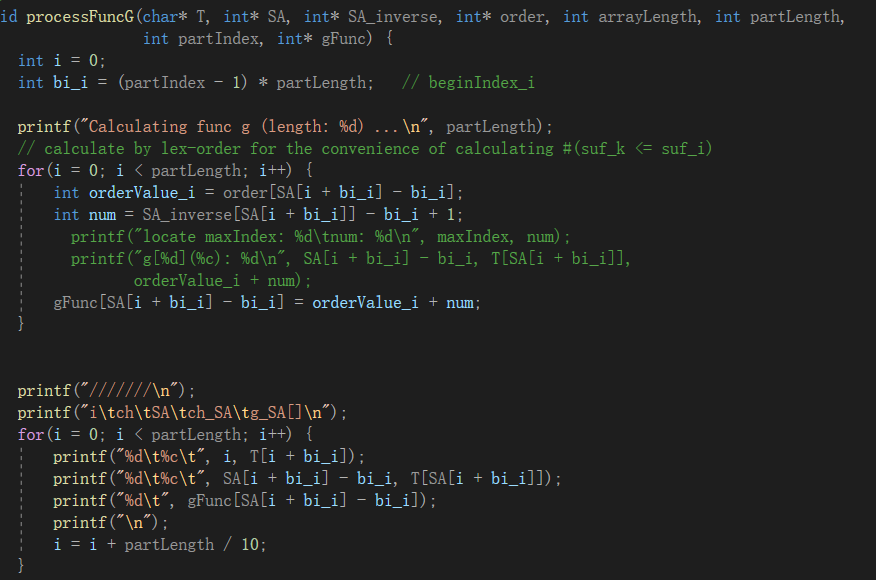
1. 对于每一个sufi，计算order（sufi，T’），即这前一段Ti的l个的后缀在T’SA中的位置



若计算order（cX，T’），找最大的b，b为处于c的SA的区间里，且b的后一位开头的后缀在T’中的字典排名是不大于order（X,T’），若b为空集，则order（cX，T’）为SA中c区间的最小序号减一（可以二分查找）

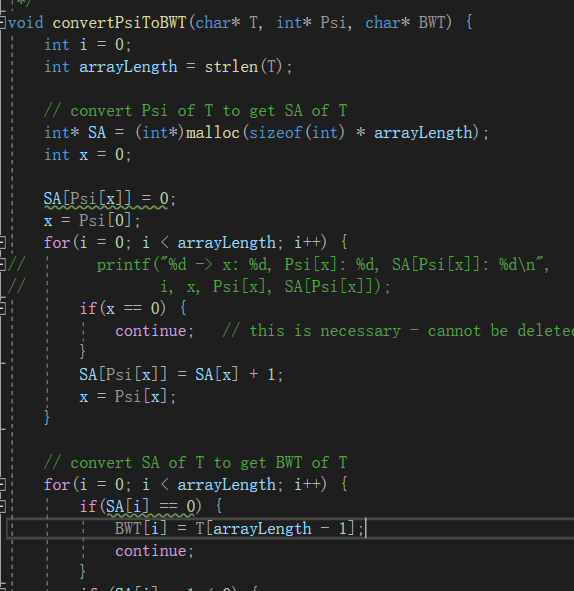
1. 对于TiT’计算ψ



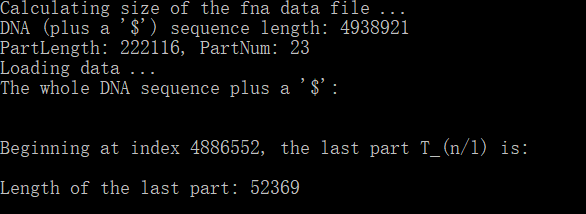


要计算f方法与g方法，分别基于计算order（sufk，T’）（在（2）中已计算），和小于suj的后缀数量

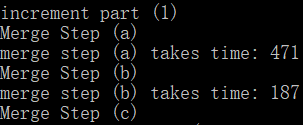
1. 通过构建好的CSA去构建BWT：将T的ψ序列转换为SA序列，再由SA序列转换为BWT



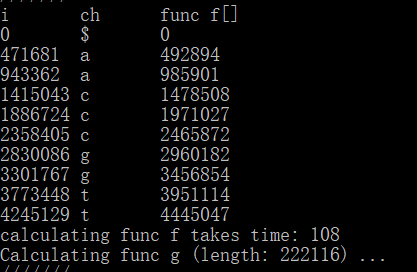
1. 测试结果及分析



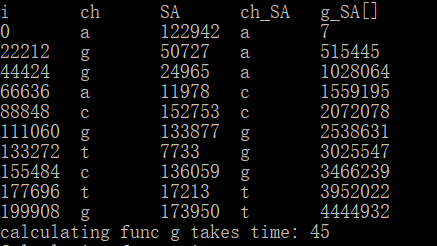
第1步，分割序列，23段，每段长222116



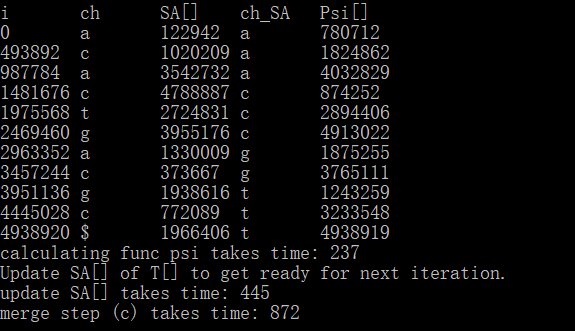
合并T1T’，即最后一次merge的a，b花费时间



最后一次f方法结果

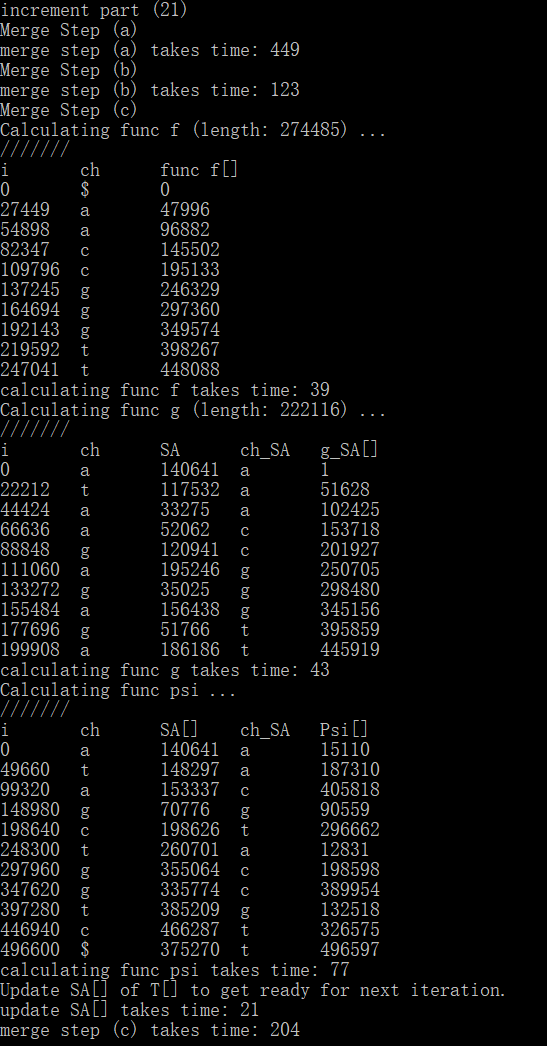


最后一次g方法结果

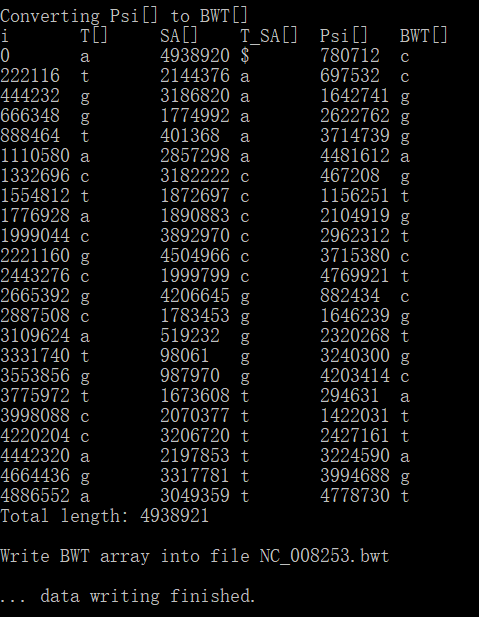


将fg合并生成ψ，可以看出ψ时间很快，而ψ转变为SA时间要花费更多，可以类比为，原始算法先算SA再算ψ

我们再与其他merge过程比较



给以看在计算T21T’，即第二次merge过程，stepa和stepb时间与最后一次大致相等，可以看出a，b两步的确只与分段长度l有关即O（llogl），而在第二步和最后一步c步骤的时间可以看出，虽然字符长度增加了20倍，但是时间花费并没有增加很大，每次涨幅平稳，基本呈线性即O（n）。在字符较少时SA与ψ的转化时间要远低于c步骤ψ方法，但是在随着字符长度越长，与l的比越大，SA与ψ的线性变换方法花费的时间，在最后一步与fgψ方法花费时间持平，并且可以预测随着数据的继续增加，fgψ的方法将快于SA，ψ线性变换方法



在最后把CSA转化为BWT方法，并且写入NC\_008253.bwt，为实验二做准备

1. 经验体会
2. 在生物信息上，随着数据量的巨大增加，不能再用上学期算法所学简单算法处理信息，O（n^2）的时间复杂度是让人算不起的时间花费
3. 通过寻找该数据结构内含的数学关系“剪去”重复操作，来优化算法，相信是以后一直会用到的思想
4. C语言的指针操作，实在是让不熟练的我很受打击，但是了解这些更接近电脑基本原理的操作，也一定会在“细节”处提升程序运行速度，还需要努力
5. 英语水平不过关，导致在第一次实验，看指导论文的时间花费不少时间
6. 附录：源代码（带注释）

https://github.com/Shaojie66/CSA.git