

# 李得草

✉ 15736818863@163.com · ☎ (+86) 157-3681-8863 · 🌐 litesla

## 🎓 教育背景

佳木斯大学, 黑龙江

2016 – 至今

在读本科生 自动化专业, 预计 2020 年 7 月毕业

## 👨‍💻 项目经历

基于哈弗曼编码的压缩解压程序

2019 年 4 月

C, Linux 个人项目

- **项目介绍:**
- **压缩流程:** 读取文件进行字节频率统计。根据每个字节的频率建立哈弗曼树, 并对每个字节进行非等长编码。读入源文件将原字节编码替换为非等长编码存入压缩文件, 并有一个标记量表示最后一个字符的有效位数。
- **解压流程:** 读取编码表, 根据编码表建立哈弗曼树。将对应的非等长编码还原为原来每个字节的编码, 存入解压文件。
- **主要数据结构和作用:**
- **Huffman tree:** 作为压缩和解压的编码和翻译的数据结构。
- **Bitqueue:** 在压缩中存储暂时存在于内存的翻译过的编码, 共有两种出队方式字符串出队和字节出队, 字符串出队的作用是用来处理队列饱和度过大时进行的压缩, 字节出队是为了存储当读取完文件时, 最后一个不满一个字节的 **有效 Byte**, 并且两个函数分别传出返回的字符串长度和字节有效位数。
- **Queue:** 用于解压当中, 将读入的字节按照二进制展开存入 **Queue** 中去, 每次循环出队字节流式处理匹配。将匹配到的字符压入文件。直到 **Queue** 当中剩余几个无效字符为止。
- **不足和改进:**
- 处理时间有待改进, 因为我是按照字节读入和存入的, 每次读入 **I/O** 都有可能将程序挂起, 浪费大量时间。可以进行块式读取 **I/O**, 减少挂起的次数。

Raspberry Pi 集群运行状态监控系统

2018 年 10 月 – 2018 年 12 月

C, Linux 个人项目

- **项目介绍:**
- 基于 **socket** 编程和 **TCP** 协议实现 **Master** 端与多台 **Client** 端之间的通信, 完成 **Master** 端对多台 **Client** 端的监控。
- **项目流程:**
- **Client 端:** 支路线程 **Client** 端运行 **Bash** 脚本收集主机状态信息, 并进行检测, 如果有报警信息直接发送给客户端, 如果没有暂存在本地。主线程发送心跳信息, 等待 **Master** 端的连接请求。
- **Master 端:** **Client** 端发送心跳信息, 当 **Master** 在线时存入链表结构, 并标记 **Hash\_table**, **Master** 端遍历链表并主动连接 **Client** 端收集信息。如果 **Master** 端没有连接到 **Client** 端的时, 删除链表节点和 **Hash\_table** 的标记。并设有警报端口, 如果有信息传入时打印在屏幕上。
- **项目亮点:**
- 运用 **Hash\_Table** 控制接受 **IP** 范围和防止重复插入, 比遍历链表的速度要快。

- 学习成果:
- 了解了蒙特卡洛树搜索的过程及其中棋盘设置和落子策略的选择以及对蒙特卡罗树的剪枝的过程
- 通过枚举类型和函数指针数组的技巧对相应的引擎进行初始化；对库函数进行二次封装以提高其健壮性；在棋局相关的内部条件判断中使用 Linux 内核中的 `likely` 和 `unlikely` 宏，提高 CPU 进行分支预测的成功率。
- Pachi 中的分布式版本主要是依靠 `distributed(Master)` 和 `UCT(Slave)` 两个引擎共同完成，Master 在每次向 Slave 发送消息前首先会收集整理共享 `tree_node` 数据，然后发送共享 `tree_node` 数据给各个 Slave，每个 Slave 各自存储一棵树并且在没有命令的时候每个 Slave 均在持续的做 `pondering`。
- 对手局中 Pachi 会“深思熟虑”，MCTS 在后台进行情况模拟，利用对手时间增加模拟次数。当对手落子后，Pachi 能在 MCT 中搜索到最有状态并将最优节点进行提根操作，回收多余节点，降低空间开销。

## 🔧 IT 技能

---

- 编程语言: 熟悉 C、C++、Bash、了解 Python.
- 平台: Linux.
- 数据结构: 基本掌握顺序表、链表、栈、队列、并查集、哈希表、堆、平衡二叉树 (AVL-Tree, SBT, RBTree) 等数据基础结构和相关操作.
- 算法: 掌握查找 (二分、三分、哈希表)、排序 (快排、归并、堆排序、桶排序)、搜索 (DFS、BFS)、简单动态规划、图论入门 (FloodFill、Prim、Kruskal、Dijkstra)、字符串匹配算法 (KMP、Sunday、Shift-And、字典树及 AC 自动机) 等基础算法.

## ♡ 获奖情况

---

二等奖, 中青杯全国大学生数学建模大赛	2018 年 4 月
一等奖, 全国大学生数学建模竞赛省级	2017 年 9 月
第五名, 第十二届中俄城市数学与程序设计竞赛	2018 年 9 月
三等奖, 第十四届黑龙江省程序大学生程序设计竞赛	2019 年 5 月

## 📄 其他

---

- 技术博客: [https://blog.csdn.net/weixin\\_38331384](https://blog.csdn.net/weixin_38331384)
- 在校期间获得过国家励志奖学金和多次校级奖学金
- 担任 2018 年黑龙江省全国青少年信息学奥林匹克竞赛 (NOIP) 夏令营助教
- 校 ACM 实验室主要负责人, 负责对低年级队员的算法与数据结构的培训、辅导