|  |  |  |
| --- | --- | --- |
| 实验1 线性表 | | |
| **实验室**：  **因疫情在家中完成** | **实验日期:**  **2020.05.06** | **周次、节次：**  **填写本次实验上课的周次等**  **第（ ）周/星期( )第( )节** |
| **班级**： **18计科四班** | **学号**： **20182714** | **姓名**： **必填** |
| **评分标准：**  难度系数  工作量  完成程度  报告质量等综合评定的成绩  **老师评语：** | | |
| **一、开发环境及工具：**   1. **实验软件系统配置：（注意标注版本号）**   操作系统：**Window10**  开发工具：**vs stdio 2019**   1. **硬件环境：**   硬件基本配置主要参数： | | |
| 1. **实验目的：** 2. **实现一元稀疏多项式的定义，初始化** 3. 熟悉链表的生成，删除插入等 4. 了解vs stdio和GitHub的使用，优化 5. malloc函数的运用 6. 链表的初始化，结构体的使用 | | |
| **三、你的选题：**  **一元稀疏多项式（单链表）**  一元稀疏多项式计算器。  可以用单链表作为存储结构，建立一元多项式，  结点按各项指数的升序排列，能够输出多项式，  输出时，先输出总项数，再输出各项系数、指数，  可以生成a和b下个多项式，并进行相加、相减，输出计算后的结果。 | | |
| **四、存储结构说明和定义：**  一元系数多项式的结构定义如下: | | |
| **五、主要算法:**   1. 主要算法的程序流程图          1. 核心函数的算法（**CTRL+左键访问，如果不行可在同一文件夹中打开**）   **●多项式插入算法**    **●多项式输出算法**    **●多项式加减算法**     1. 其他子函数的函数头     **此处给出程序中所有的函数头，如果有参数请加以说明** | | |
| **六、程序运行结果测试：**  **录屏演示** | | |
| 1. **本次实验小结：** 2. 分析自己的主要算法   **算法采用链表的形式，链表的优点是：在插入和删除操作时，只需要修改被删节点上一节点的链接地址**  **没有顺序存储结构中的插入和删除操作需要移动大量元素的缺点但是失去了顺序存储结构随机存取的特性。**  **在算法中，一开始准备随机插入，最后再写一个排序算法，后来觉得后续排序算法过于繁琐，不如在插入**  **是就选择至对应的排序，后续如果需要逆序可通过栈或者其他算法实现.**  **对于插入算法的时间复杂度，有数据和对应要插入的多项式组有关，假设原多项式组有n个元素，平均时**  **间复杂度为O（n）.**   1. 说明本次实验中遇到的问题和你的解决方案   **写代码5分钟，调制编程软件2小时.在初步编写完代码后，编译器一直出现问题（具体视屏中讲明）。后**  **来一步步查资料，结果编译器问题，栽根据结果一步步改代码，优化代码就很舒服。同时学会了讲程序代**  **码挂载在网络上。程序有一个定义文件，一个声名文件，一个主函数，显得十分的简介明了.**   1. 对本次实验的完成情况自我评价   **总体不错，细节还可以继续优化，如判断内存满的语句等等** | | |
| 1. **附录： （源程序代码）**   **代码在同一文件夹内，也可点击链接进入GitHub仓库查询**  <https://github.com/sunbaocai/Data-structure-test2.0/tree/master/%E6%95%B0%E6%8D%AE%E7%BB%93%E6%9E%84%E8%AF%95%E9%AA%8C/%E7%AC%AC%E4%B8%80%E7%AB%A0%E7%BA%BF%E6%80%A7%E8%A1%A8/%E7%A8%80%E7%96%8F%E5%A4%9A%E9%A1%B9%E5%BC%8F> | | |
| **九、你使用或推荐的参考资料：**  **引用和指针的区别：**<https://blog.csdn.net/listening_music/article/details/6921608>  **Vs stdio 2019的使用：**<https://blog.csdn.net/qq_36556893/article/details/88603729>  **Github desktop的使用：“B站搜索第一个”** | | |