



# 面向测控专业的非线性数据结构算法 实验设计

黄敏, 苟轩\*, 刘科

(电子科技大学自动化工程学院, 成都 611731)

**摘要:** 数据结构是软件设计领域的重要课程, 对于测控专业的学生来说, 掌握数据结构知识尤为重要。然而, 在教学过程中, 如何将通用的数据结构知识与测控专业面临的实际工程问题相结合, 以及如何选择恰当的非线性结构问题案例, 以便学生在知识理解和实践应用两个层面加深对课程内容的掌握, 是教学中需要解决的关键问题。为此, 在教学实践中, 以测控领域科研中常用的自动测试命令解析程序设计为例, 设计了非线性数据结构算法实验。通过科教融合的方式拓展教学的深度和广度, 使学生能够将理论知识与实践技能相结合, 有效提升他们解决实际工程问题的能力。

**关键词:** 数据结构; 自动测试; 二叉树; 遍历; 命令解析

中图分类号: G642.4

文献标志码: A

DOI: 10.12179/1672-4550.20230442

## Experimental Design of Nonlinear Data Structure Algorithm for Measurement and Control Majors

HUANG Min, GOU Xuan\*, LIU Ke

(School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China)

**Abstract:** "Data structure" is an important course in the field of software design, and it is particularly important for students majoring in measurement and control technology to master the knowledge of data structure. However, in the teaching process, how to combine the general data structure knowledge with the practical engineering problems faced by the measurement and control technology major, and how to choose the appropriate nonlinear structure problem cases, so that students can deepen the grasp of the course content in both knowledge understanding and practical application, are the key problems to be solved in the teaching. Therefore, in the teaching practice, taking the automatic test command analysis program commonly used in the field of measurement and control technology research as an example, the nonlinear data structure algorithm experiment is designed. Through the integration of science and education, the depth and breadth of teaching are expanded, so that students can combine theoretical knowledge with practical skills, and effectively improve their ability to solve practical engineering problems.

**Key words:** data structure; automatic testing; binary tree; traversal; command parsing

数据结构课程是计算机科学与技术专业必修的核心课, 同时也是工科电子类专业的重要课程。目前国内的测控技术与仪器专业较少开设有关数据结构方面的课程, 但测控专业在测控系统软件设计时必不可少要用到数据结构的知识。

测控技术与仪器专业的学生通过软件技术基础——数据结构课程的学习, 能够解决简单线性

问题的程序设计; 但是面对非线性问题, 学生往往束手无策, 只能寄希望于用线性结构算法设计解决非线性结构问题。因此, 学生很有必要掌握解决非线性结构问题的算法设计。

软件技术基础——数据结构课程内容结构比较清晰, 该课程从实际问题出发, 给出问题的数学模型, 并在此基础上建立物理模型, 从而进一

收稿日期: 2023-09-19; 修回日期: 2023-11-15

基金项目: 电子科技大学新生项目课程建设项目(2023PBLF089); 电子科技大学挑战性课程建设项目(2022CLC-168)。

作者简介: 黄敏(1983-), 女, 硕士, 实验师, 主要从事宽带时域测试技术及仪器、集成电路测试等方面的研究。

\*通信作者: 苟轩(1986-), 男, 硕士, 实验师, 主要从事精密功率测量和智能传感方面的研究。E-mail: gouxuan@uestc.edu.cn

步确定计算机中解决该问题所用的数据存储结构及算法实现。虽然课程内容结构清晰,但学生学习此课程有很大的难度,难点在非线性的存储结构和算法<sup>[1]</sup>。教材上的非线性结构举例没有与实际工程问题关联,教师课堂上给出的示例也仅来源于生活:如非线性结构中的树状结构一般以族谱为例,或者文件系统的目录结构为例,这些示例不利于学生将所学课程内容与后续专业课程学习、科研工作相结合。为此,课程组老师通过探索测控技术与仪器专业相关研究领域:发现测控专业在自动测试系统中用到的可编程仪器标准命令(standard commands for programmable instruments, SCPI)的命令集是这一难点的典型实际工程应用案例。因此可将该内容设计为测控技术与仪器专业的非线性结构算法实验,有助于学生将所学的软件知识应用到测控专业实际的科研工作中,并运用所学数据结构相关知识解决实际工程领域中的复杂问题<sup>[2]</sup>。

## 1 实验背景

信息时代,小到芯片大到飞机,都离不开测试,由于测试条件的限制,大量测试都通过自动测试实现。自动测试的实现依赖于计算机软硬件技术、测试控制技术等。要实现自动测试,硬件上离不开计算机和测试仪器(可编程仪器)<sup>[3]</sup>。要实现通过计算机控制测试仪器完成特定的测试任务,计算机与测试仪器之间必须通信,如何通信?在自动测试中采用了与自然语言相似的形式,通过文本命令在计算机与仪器间进行通信,实现测试功能<sup>[4]</sup>。如图1所示,计算机控制示波器完成电压峰-峰值测试,计算机向示波器发送命令“:MEASure:VPP?”,示波器接收命令后执行测量操作,将测得电压值返回给计算机<sup>[5]</sup>。由于计算机和可编程仪器只能处理0|1二进制数据,因此为了支持程控命令,在计算机和可编程仪器中必须有命令解析程序将文本命令转换为0|1形式的命令码。

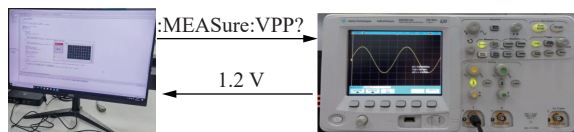


图1 自动测试过程示意图

计算机和可编程仪器间传递的程控命令并不是任意的,标准化组织对命令做了统一,即SCPI。可编程仪器根据自身功能,从SCPI命令集中选择命令形成自己的可编程命令集<sup>[6]</sup>。SCPI标准中对命令进行大类划分,各大类下又进行子功能划分,各子功能还可以进一步划分,直到划分为能够由仪器完成的最小功能,因此SCPI命令是一种一对多的非线性结构。在可编程仪器设计中,命令解析是软件设计的一个重要组成部分。培养具有测控仪器与系统设计能力的工程技术人员是测控技术与仪器专业的培养目标之一,因此选择SCPI命令解析作为软件技术基础——数据结构课程中非线性结构的实验,既可以帮助学生掌握非线性结构的算法设计,又可以帮助学生理解自动测试的实现原理,从而使学生能更快适应后续测控仪器与系统的实践学习<sup>[7]</sup>。

## 2 实验设计

在本文的实验设计中,先让学生了解实验目的;再进行问题分析,建立数学模型;再运用数据结构中典型二叉树结构完成相关算法<sup>[8]</sup>;最后学生使用框架程序对算法进行测试。该实验能有效提升测控专业学生在测控工程中的实践能力和计算机知识的运用能力<sup>[9]</sup>。具体实验过程分为以下4个阶段。

### 2.1 理解实验目的

SCPI程控消息(也称为程控命令)包括命令头和参数,程控消息如:MEASure:VPP? CH1,其中:EA-Sure:VPP?为程控命令题头,CH1为参数<sup>[10]</sup>。一条SCPI命令可以由一个或多个关键字构成,在:MEASure:VPP? CH1命令中包含2个关键字MEASure和VPP。SCPI命令是一种层级结构,又称之为树状结构。根节点代表某一类特定功能,根节点下再针对不同子功能划分为不同的孩子节点,子功能还可以再进一步细分,最后一层为执行某一特定功能的叶子节点。控制仪器执行某一功能的命令必须指定从根到叶子节点的全路径。下面以参数测量命令树为例(为了简化起见,只选取整个参数测量的一部分功能)对命令树的构成加以说明。简化后的参数测量命令树结构如图2所示,其根节点为MEASure,代表与参数测量相关的所有命令都在该命令树下,该树下又分为4个子功能,分别是频率测量(FREQ?)、峰-峰值测量(VPP?)、相位

测量(PHAS?)和统计运算(STAT), 统计运算命令下又设复位(RES?)和计数值增加(INCR)子功能。

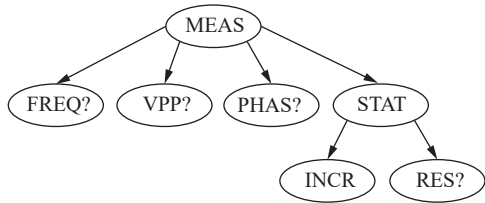


图 2 简化后参数测量命令树结构

由于仪器不是单一功能的设备, 不同类别的仪器有各自的命令集<sup>[10]</sup>, 因此每类仪器的命令集由多棵命令树构成, 这些命令树共同构成仪器的命令森林。测控领域常用仪器示波器的程控命令集片段如表 1 所示。

根据该命令集片段, 可得到对应的命令树如图 3 所示, 该命令集包含 3 棵命令树, 分别为 CHANNEL 命令树, TIMEbase 命令树和 MEASure 命令树。

表 1 示波器命令片段对应的命令树森林

命令集	命令	说明
CHANNEL 通道	:CHANnel<n>:COUPling <coupling>	耦合方式
	:CHANnel<n>:PROBe:STYPe <signal type>	探头类型
	:CHANnel<n>:SCALe <scale>[<suffix>]	垂直刻度
	:CHANnel<n>:OFFSet <offset> [<suffix>]	直流偏移
TIMEbase 时基	:TIMEbase:SCALe <scale_value>	时基刻度
	:TIMEbase:REFerence <reference>	触发参考点
	:TIMEbase:WINDow:SCALe <scale_value>	延迟扫描的刻度
	:TIMEbase:WINDow:POSition <pos value>	延迟扫描的位置
MEASure 测量	:MEASure:FREQuency? [<source>]	测量频率值
	:MEASure:VPP? [<source>]	测量峰-峰值
	:MEASure:PHASe? [<source1>][,<source2>]	测量相位
	:MEASure:STATistics:RESet?	测量统计运算参数复位
	:MEASure:STATistics:INCRement	测量统计运算数值增加

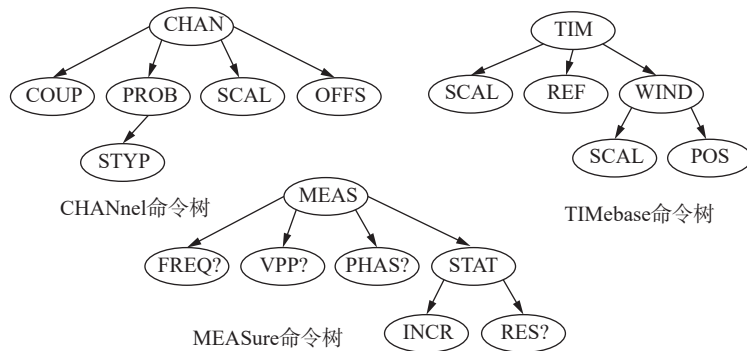


图 3 示波器命令片段对应的命令树森林

在自动测试中, 用户根据要实现的功能选择相应的命令, 通过测试程序将命令对应的程控消息发送给仪器, 仪器接收到程控消息后, 对程控消息进行命令题头和参数的分离, 基于命令题头对命令树进行遍历, 遍历的过程可确定是否有该命令, 如果命令存在, 在最终的叶子节点处可以得到对应的操作内码, 最后将操作内码和参数交给仪器执行。

### 2.2 分析实际问题

通过上述分析, 命令解析要完成的操作包括: 命令树的建立、程控消息题头和参数的分离、命令树遍历得到操作码、参数处理、交付操作内码和参数。其中命令树建立和命令树遍历得到操作码可利用软件技术基础——数据结构课程中非线性结构的相关算法实现。

通过上述分析得到程控命令解析问题的数学

模型为层次模型, 可以在实验中完成的操作包括命令树构造, 命令树遍历解析得到操作码。

### 2.3 解决实际问题

在软件技术基础——数据结构课程中采用二叉树处理层次数学模型的问题, 对应的物理模型为二叉树模型。上述分析中命令集对应的树状结构是包含多棵树的命令树森林, 如图 3 所示, 要得到二叉树首先必须将每一棵命令树转换为二叉树, 其次再将二叉树森林转换为二叉树。命令树转换为二叉树的基本规则: 根节点的第一个孩子节点作为根节点的左孩子, 第一个孩子节点的兄弟节点作为第一个孩子节点的右孩子节点, 兄弟节点成为孩子节点的右链, 以此类推, 节点都做相似变换。命令树森林中每棵树都变为二叉树, 在这些二叉树中, 树的根节点只有左链。将每棵二叉树进行右链连接, 命令树森林变成一棵二叉树<sup>[11-12]</sup>。基于图 3 的命令树森林得到的二叉树如图 4 所示。

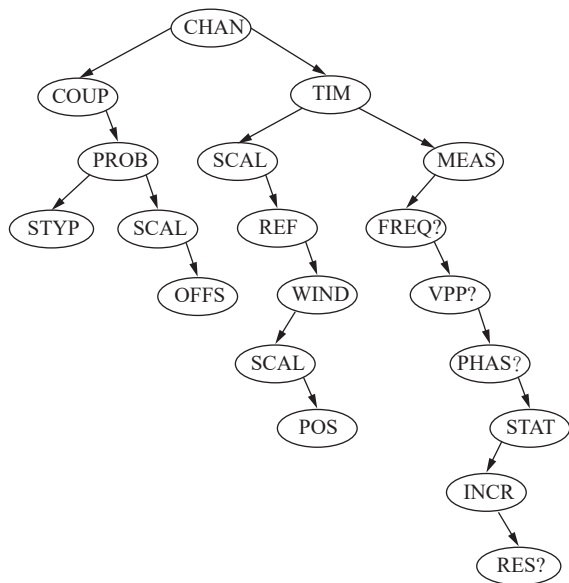


图 4 示波器命令片段对应二叉树

由于命令集中包含的命令数目是动态的, 因此命令二叉树采用链式结构动态生成, 基于二叉链表的命令解析问题所对应的物理模型, 其抽象数据类型定义如图 5 所示。在该抽象数据类型中包含数据成员二叉树根节点 root, 二叉树节点类型 BiTnode 在具体实现中给出定义, 该节点定义需包含数据域, 指向左右孩子节点的指针域。基本算法包括创建二叉树算法和遍历二叉树算法, 在创建二叉树 CreateTree( BiTnode\*\*proot)算法中, 根据仪器程控命令文本文件动态生成二叉树, 树的根节点为算法输入参数 proot 所指向的节点。遍历二叉树算法完成对命令消息的解析, 根据命令消息中的关键字对二叉树进行访问。如果访问成功, 则得到命令执行的代码; 如果访问失败, 则退出。

```

ADT CmdTree
{
    BiTnode root; //树的根节点
    CreateTree(BiTnode**proot); //创建二叉树
    TraverseTree(BiTnode *proot, char* cmdstr); //遍历二叉树
}

```

图 5 命令解析抽象数据类型定义

通过上述对命令解析问题的分析, 得到该问题的数学模型和物理模型, 并给出该问题的抽象数据类型定义。测控技术与仪器专业非线性数据结构实验进入实现阶段, 在该阶段中需完成如下工作: 命令解析问题中节点数据类型定义, 二叉链表节点定义, 根据仪器命令文件创建命令二叉树算法实现, 根据命令消息对二叉树遍历得到执行代码算法实现, 验证算法正确性的测试程序。

二叉树中处理问题的数据类型定义如图 6(a) 所示, 实验中为了突出教学重点, 将实验的核心落在二叉树对命令的解析部分, 对于参数的处理由框架程序完成。二叉树中二叉链表的表节点定义如图 6(b)所示。

```

typedef struct
{
    int code; //执行内码
    int idefault; //缺省节点标记
    char cmdname[40]; //节点命令串
    ScpiParaList para; //支持参数类型
}ScpiTreeData;

```

(a) 二叉树中节点数据类型定义

```

typedef struct
{
    ScpiTreeData data; //节点数据域
    struct scpiBTnode * LC; //左孩子节点
    struct scpiBTnode * RC; //右孩子节点
}ScpiBTNode;

```

(b) 二叉链表节点类型定义

图 6 二叉树相关数据类型定义

在命令解析中，与二叉树相关的算法定义如图 7 所示，CreatCmdTree 算法根据仪器命令集创建二叉树；ParserProc 算法，对程控消息在二叉树中进行遍历、解析，成功则返回执行内码；Preorder-Visit 算法，对二叉树进行先序遍历，打印出遍历信息，可用于检验创建的命令二叉树的正确性。

```
int CreatCmdTree(ScpiBTNode ** prootnode);
int ParserProc(ScpiBTNode * prootnode, char *cmdmessage);
void PreorderVisit(ScpiBTNode * prootnode);
```

图 7 二叉树相关算法

仪器根据自身的功能选择支持的程控命令，所有命令保存到文本文件 commandtree.txt 中，算法 CreatCmdTree 打开程控命令文件，逐一读出

每条程控命令，进行命令题头和参数的分割；再对命令题头的不同部分进行分割，得到不同层级的关键字，将关键字保存到命令队列中；从队列中依次取出命令关键字，对每个关键字进行二叉树遍历。在二叉树遍历中，如果遍历到空分支仍没有找到该关键字，则构造该关键字节点，插入到二叉树的当前位置；如果在遍历过程中找到关键字节点，则说明该节点在命令树中已存在，当前节点为该节点的左分支，从命令队列中取下一级命令，重复前面的遍历查找插入工作，直到队列为空，算法流程如图 8 所示。图 8(a)为生成命令二叉树的总流程，图中 A 表示一条程控命令构造流程；图 8(b)为 A 的流程图。

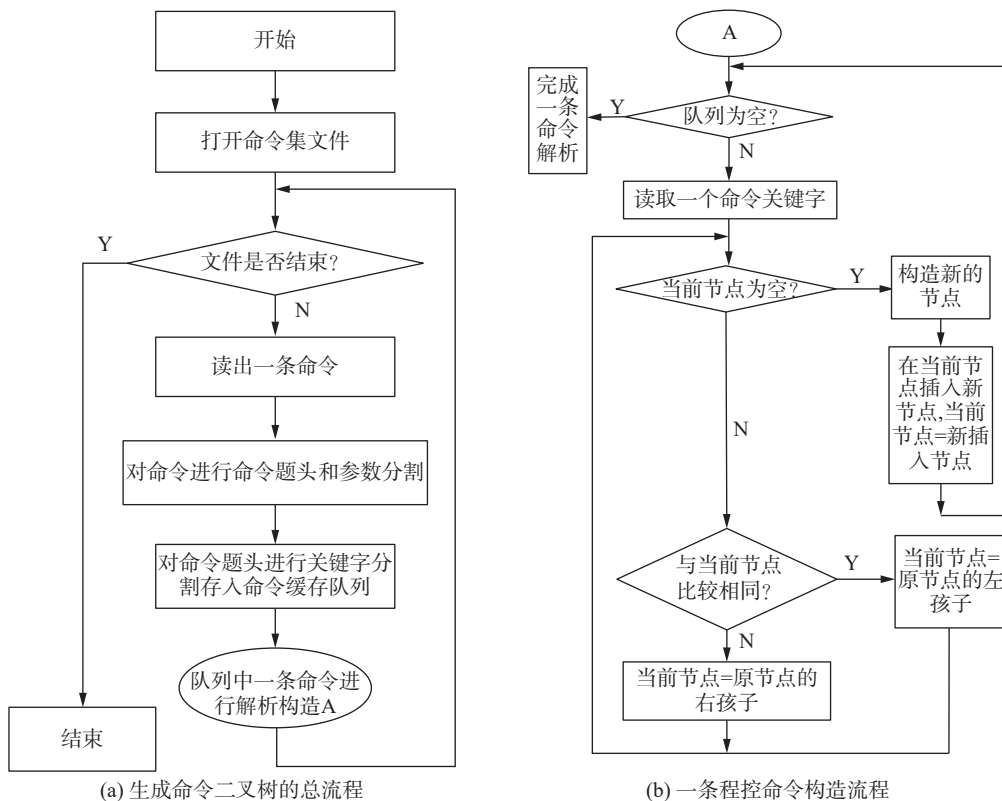


图 8 二叉树的算法流程

命令解析对收到的程控命令进行解析得到仪器能够理解的执行内码，将命令内码和参数传递给仪器的器件功能执行。在本实验中突出非线性数据结构的处理，参数处理部分代码由程序框架提供，不属于实验要求部分。算法 ParserProc 有两个输入参数，参数 prootnode 为命令树根节点，参数 cmdmessage 为程控消息命令题头。在算法中，首先对命令题头进行分割，分割后的各级命令关

键字存入队列；取出队列中的关键字，从根节点开始对命令树进行右链遍历。如果找到相同的节点且队列中没有下一级命令，则当前节点为最终查找的终端节点；如果队列中有下一级命令，则取该节点的左分支为当前查找节点，重复上面的关键字查找。命令解析算法流程图如图 9 所示，图中 pcur 为节点指针，保存查找中的当前节点地址。

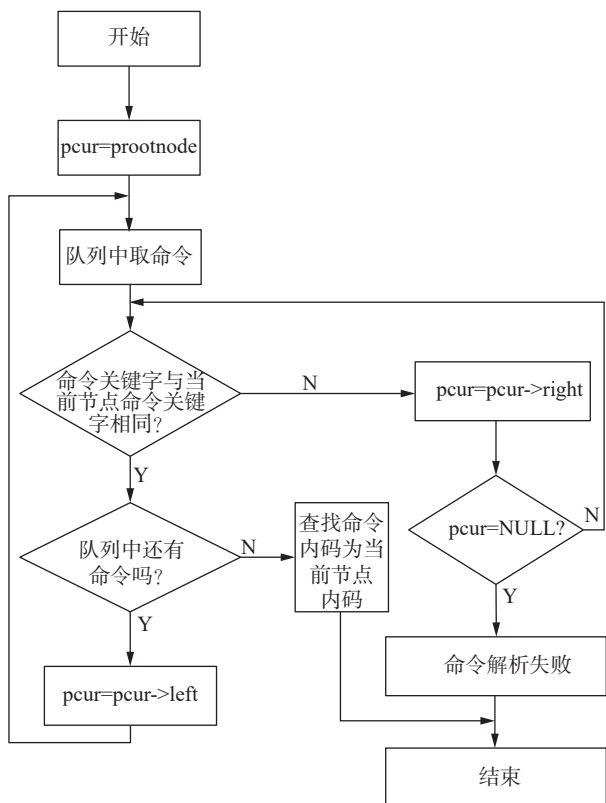


图 9 命令解析算法流程图

### 2.4 算法验证

为了验证命令树构造的正确性, 在实验中可任选二叉树遍历 3 种算法(先序遍历、中序遍历和

后序遍历)<sup>[13-14]</sup>中的一种, 对构造的命令二叉树进行遍历, 并在屏幕中输出遍历后得到的命令树。本实验算法 PreorderVisit 输入参数 prootnode 为命令树根节点, 该算法通过先序遍历规则, 对算法 CreatCmdTree 构造的二叉树进行访问, 通过屏幕输出二叉树, 其结果如图 10 所示, 点击按钮载入命令集并构造命令树, 选择遍历算法, 启动对命令二叉树的遍历, 遍历结果显示在图中区域 2。区域 2 中双竖线代表一棵树的根节点, 3 个方框里分别对应三棵命令树的先序遍历序列, 对照区域 1 和区域 2, 证明得到的命令树完全正确。由于二叉树遍历算法较为成熟, 算法流程<sup>[8]</sup>在这里略去。

针对不同能力的学生, 非线性数据结构的实验进行了分层次设计<sup>[15]</sup>, 如图 11 所示。能力较弱的学生借助该实验可以学习非线性结构相关算法如何解决实际问题(图中给出先序遍历算法实现代码); 能力较强的学生, 可以挑战设计非线性数据结构相关算法, 提高算法设计能力, 中序和后序遍历算法可留给学生自行尝试设计。

本实验开设之前, 学生反映不知道如何应用树状结构; 开设该实验后, 学生不仅学习了树状结构算法设计, 还学习了测控专业的仪器程控实验原理, 同时在算法设计的基础上进一步学习了图形化界面应用程序设计, 收获很大。学生使用后的评教情况如表 2 所示。

非线性结构—SCPI命令二叉树实验

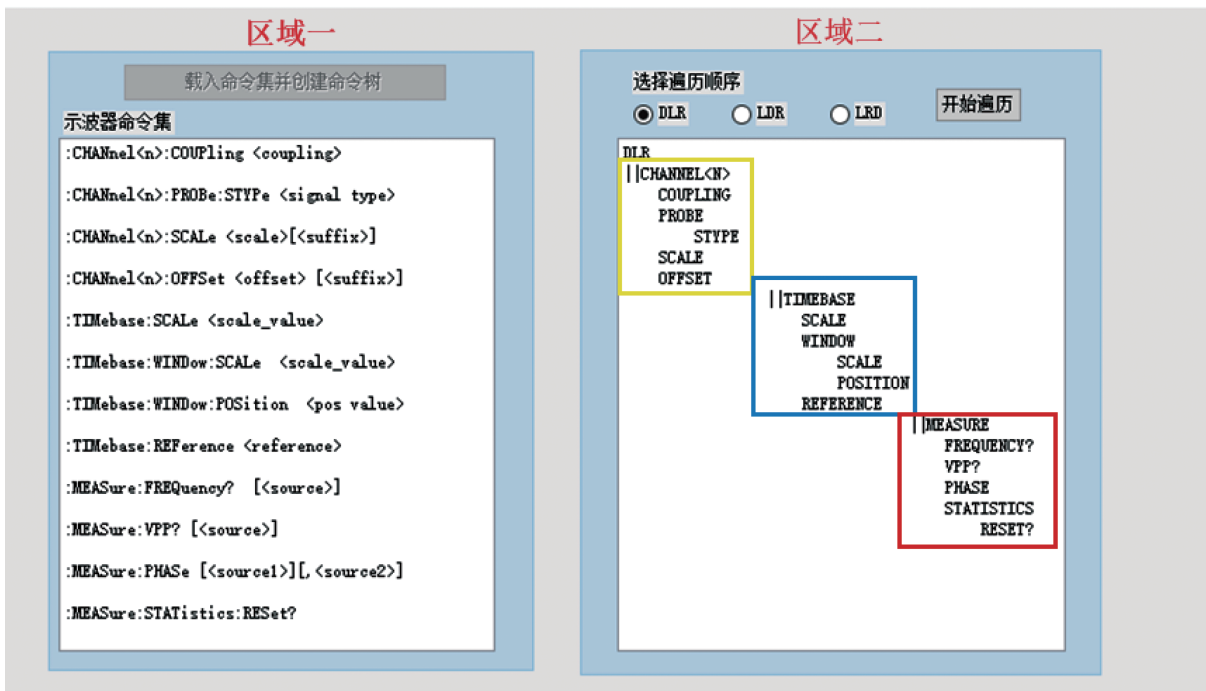


图 10 先序遍历算法访问命令二叉树

```

/*****
/*Name   : PreorderVisit   */
/*Func   : 先序遍历二叉树, */
/*       : 用于验证二叉树是否正确 */
/*Input  : None           */
/*Output : None           */
/*Note   : None           */
/*****
void PreorderVisit(ScpiBTNode *pnode)
{
    static level=0;
    static right=0;
    static left = 0;
    static root = 0;
    ScpiBTNode *pcurnode=pnode;
    if(pcurnode==NULL)
    else
    {
        for(int i=0;i<(root*4+left);i++)
        if(left==0)
            #ifdef PRINTF
            #else
            #endif
            #ifdef PRINTF
            #else
            #endif
            if(pcurnode->LC!=NULL)
            {
                left++;
                preorder(pcurnode->LC);
                left--;
            }
            if(left==0) root++;
            if(pcurnode->RC!=NULL)
            {
                right++;
                preorder(pcurnode->RC);
                right--;
            }
        }
    }
}

/*****
/*Name   : InorderVisit   */
/*Func   : 中序遍历二叉树, */
/*       : 用于验证二叉树是否正确 */
/*Input  : None           */
/*Output : None           */
/*Note   : None           */
/*****
void InorderVisit(ScpiBTNode *pnode)
{
}

/*****
/*Name   : PostorderVisit */
/*Func   : 后序遍历二叉树, */
/*       : 用于验证二叉树是否正确 */
/*Input  : None           */
/*Output : None           */
/*Note   : None           */
/*****
void PostorderVisit(ScpiBTNode *pnode)
{
}
    
```

图 11 3 种顺序遍历算法

表 2 课程评教情况

课程代码	课程名称	文字评教内容
F0600420	软件技术基础 ——数据结构	教学设计由浅入深，循序渐进，学生实际操作机会多 老师讲解由浅入深，条理清晰，重点突出 老师很严格，作业很多，还有很多实验，还有与科研结合的实验，而且还是非线性结构的.....但你确实可以学到东西 内容充实、有思想性和深度，课堂教学信息量大 老师课堂生动，能学到很多工程上的经验 以前学C语言的实验都是基于命令行的，通过本课程实验学习了图形化的应用程序设计 课程难度还是有点大，特别是到了非线性结构部分，简直“晕菜”，但是通过与科研课题结合的实验，让我第一次感觉到自己所学的东西真的可以拿来用，还是有点激动

### 3 结束语

数据结构之非线性树状结构实验，在以往教学中，仅通过教材给定的基于一组数据构造二叉排序树来完成，没有与实际科研项目相结合，不具有工程实操性。

新设计的 SCPI 命令解析非线性数据结构实验，与测控专业的科研项目紧密结合，有效提升学生运用数据结构课程知识解决实际工程问题的能力。

### 参考文献

- [1] 连远锋, 张鑫, 庄永琪, 等. 数据结构与算法可视化调试虚拟仿真实验系统设计[J]. 实验技术与管理, 2023, 40(5): 122-129.
- [2] 于真. 基于OBE理念的数据结构与算法课程教学改革研究[J]. 科技风, 2023(19): 123-125.
- [3] 赖小红, 金卫. 疫情下自动测试系统课程实验教学设计与实施[J]. 实验科学与技术, 2022, 20(2): 62-66.
- [4] 李庆钰, 翟柳, 李贞, 等. 基于命令字驱动的自动化测试框架研究[J]. 现代信息科技, 2022, 6(12): 86-88.

(下转第 75 页)