



2013 COMPUTER

计算机教学研究与实践

—— 2013学术年会论文集

◎ 浙江省高校计算机教学研究会 编



ZHEJIANG UNIVERSITY PRESS
浙江大学出版社

013066711

TP3-53
24
2013

计算机教学研究与实践

——2013 学年会论文集

浙江省高校计算机教学研究会 编



ZHEJIANG UNIVERSITY PRESS
浙江大学出版社



北航

C1674636

TP3-53

24

2013

图书在版编目 (CIP) 数据

计算机教学研究与实践: 2013 学术年会论文集 / 浙江省高校计算机教学研究会编. —杭州: 浙江大学出版社, 2013. 8

ISBN 978-7-308-11931-3

I. ①计… II. ①浙… III. ①电子计算机—教学研究—高等学校—学术会议—文集 IV. ①TP3-42

中国版本图书馆 CIP 数据核字 (2013) 第 175566 号

计算机教学研究与实践——2013 学术年会论文集

浙江省高校计算机教学研究会 编

责任编辑 吴昌雷 黄娟琴

封面设计 刘依群

出版发行 浙江大学出版社

(杭州市天目山路 148 号 邮政编码 310007)

(网址: <http://www.zjupress.com>)

排 版 杭州中大图文设计有限公司

印 刷 杭州日报报业集团盛元印务有限公司

开 本 787mm×1092mm 1/16

印 张 10.25

字 数 249 千

版 印 次 2013 年 8 月第 1 版 2013 年 8 月第 1 次印刷

书 号 ISBN 978-7-308-11931-3

定 价 35.00 元

版权所有 翻印必究 印装差错 负责调换

浙江大学出版社发行部联系方式: 0571-88925591; <http://zjdxcbbs.tmall.com>

目 录

专业建设与课程体系建设

(以姓氏拼音为序)

基于计算思维的程序设计课程教学实践——基本程序设计教学示例	杜建生(3)
农业信息化领域教学改革与实践	顾沈明 吴伟志 吴远红(12)
创新创业教育与高职计算机专业教育体系的相融性研究	贾淑红(15)
网络设计与工程课程实验设计与实践	蒋巍巍 王海舜(19)
基于“三阶递进校企合作”的高职动漫专业实践教学体系探析	金斌英 阮 威(23)
本科数据库课程教学改革的研究与探索	陆慧娟 徐展翼 高波涌 何灵敏(28)
以项目实践为主导的计算机实训课程探讨	马 亮 王海舜 李文胜(33)
大学计算机课程中的计算思维培养方法探索	谢红霞 钟晴江(37)
基于 CDIO 模式的高职软件专业改革研究与实践	许益成 陈丽婷 闻红华 周 丹(41)
以职业能力为导向的数字媒体技术专业平面设计模块课程体系构建	应 英(47)

课程建设

(以姓氏拼音为序)

高职 Java 语言程序设计课程教学改革探讨	冯姚震 余先虎 黄 文(59)
基于增量模型的“数据结构”实验教学体系研究	赖 梅(63)
高职艺术设计中澳合作项目课程开发研究——以网页设计与制作课程为例	李振华(67)
基于个体差异的分类实训项目研究与探讨——以三维图形设计实训课程为例	钱燕婷 代绍庆 李 平(73)
基于 CDIO 理念的高职数字图形图像设计与处理课程改革探索及实践	阮 威 金斌英(78)

管理信息系统设计与课程实践的探讨	王竹云(83)
面向计算思维能力培养的程序设计课程的改革和实践	张广群 汪杭军 尹建新(88)

教学方法与教学环境建设

(以姓氏拼音为序)

软件开发类毕业论文内容的基本要求	安立新 徐展翼(95)
“数据结构”实践性考核方法探讨	陈志杨 雷艳静(99)
高职计算机基础公共课教学内容的“同”与“不同”	韩红光(103)
以学科竞赛为驱动的大学生实践能力培养模式	韩建民 赵建民 陈中育 叶荣华 郑忠龙(107)
应用愉快教育的离散数学教学实践	胡亚红(113)
军队院校计算机实验课的探索与研究	李伟良 盛 晨(116)
基于核心编程的教学实践探索	罗国明(120)
基于自主探究与团队学习相结合的教学设计——以“遮罩动画制作”为例	王 涛 陈文青(123)
浅谈基于 RCT 的大学计算机基础类课程改革	徐恩友 韩建平 朱娅妮 洪道平(129)
以提升内在素质为核心的开放式快乐教学范式改革	应小凡 马雪英(134)
也谈 VB 程序设计语言的教学改革	于 莉(139)

实验室建设与网络辅助教学

(以姓氏拼音为序)

高职图形图像类课程形成性评价的策略和方法	金智鹏 代绍庆(145)
基于网络平台的高职计算机类毕业设计课程教学模式改革	卢彰诚(151)
实验教学示范中心开放服务平台建设	徐 卫 陈 琦(156)

专业建设与课程体系建设

基于计算思维的程序设计课程教学实践

——基本程序设计教学示例

杜建生

浙江大学城市学院计算分院, 浙江杭州, 310015

摘要: 程序设计课程是高校计算机专业学生的重要基础课程,也是非计算机专业学生学习应用计算机的必修基础课程。通过该课程的学习,培养大学生的逻辑思维、编程思想,从而提高对“计算思维”的认识。学习程序设计课程对学生而言,由于思维的转变,学习起来有一定的难度;对教师而言,如何生动进行有效的教学,设计好课程教案,将“计算思维”贯穿于整个教学过程中,让学生循序渐进地学会掌握程序设计基本内容,从一般的思维方式转变到计算思维方式,从而达到程序设计课程的教学目标。本文就如何以计算思维理论指导程序设计教学作了教学尝试和实践,与大家共同探讨和分享。

关键词: 计算思维;程序设计;教学实践;教学示例

1 引言

近几年来,计算思维的理论在高校计算机教学研究中形成热点。高校计算机基础教学发展战略思想的核心是:需要把培养学生的“计算思维”能力作为计算机基础教学的核心任务。

程序设计课程是高校计算机专业学生的重要基础课程,也是非计算机专业学生学习应用计算机的必修基础课程。通过该课程,使学生掌握程序设计课程的基本知识、基本方法、结构化程序设计和基本算法,并培养学生利用计算机解决问题的意识、方法和能力,具备利用计算机求解实际问题的基本技能,能灵活应用程序语言结合本专业知识进行程序设计,为计算机在各专业中的应用奠定基础。“程序设计”是学生最容易理解计算机求解问题的特点与方法的课程,如何在课程中体现计算思维思想和理论,培养学生的计算思维能力,让学生从朴素的自然思维转变到计算思维,是我们在课程教学方法需要思考的。我的实践是将培养大学生的计算思维作为程序设计课程的教学重点,贯穿整个教学过程,相应地调整了教学内容、教学进度和教学案例,从而提升了程序设计课程的教学效果。

2 确立程序设计中计算思维的概念

程序设计课程是高校计算机科学技术、电子信息工程和信息管理等专业一年级新生的

基础课程,也是非计算机专业学生一年级下半学期的课程。多数学生对程序设计是陌生的,刚开始学习时,注意力往往集中在程序设计语言本身,学得很勉强,找不到感觉,普遍觉得难学。

针对这一普遍情况,我们在程序设计的第一次导论课中,从计算思维讲起,讲计算机能做什么、不能做什么、又是怎么做的。从而把学习程序设计课程中最应关注和最受益的思维锻炼过程明白告诉学生,向学生说明学习程序设计课要关注大脑的思维过程,要思考计算机运行程序的思想,形成计算思维,最终达到思维的锻炼。我们通过下列求解平均值的同一问题将程序设计的基本内容联系起来,让学生去学习训练、操作实践、检验测试,一步步地掌握基本程序设计的内容并培养计算机的编程思想,逐步形成计算思维的概念。

VB 程序设计课程的教案示例如下:

- (1)从键盘输入 3 个数,求它们的平均值;
- (2)从键盘输入 30 个数,求它们的平均值;
- (3)从键盘输入 n 个数,求它们的平均值;
- (4)从键盘输入任意一批数,求它们的平均值;
- (5)从键盘输入一批数,放入数组中,然后求它们的平均值;
- (6)将求平均值的问题用函数模块 function 来实现;
- (7)将求平均值的问题用过程模块 Sub 来实现;
- (8)从数据文件中读入一批数,求它们的平均值。

C 或 C++ 程序设计课程教案示例如下:

- (1)从键盘输入 3 个数,求它们的平均值;
- (2)从键盘输入 30 个数,求它们的平均值;
- (3)从键盘输入 n 个数,求它们的平均值;
- (4)从键盘输入任意一批数,求它们的平均值;
- (5)从键盘输入一批数,放入数组中,然后求它们的平均值;
- (6)将求平均值的问题用自定义函数来实现;
- (7)利用指针操作求一批数的平均值;
- (8)从数据文件中读入一批数,求它们的平均值;
- (9)利用结构体存放一批数据,求它们的平均值;
- (10)建立一个类存放一批数据,求它们的平均值。

然后,告诉学生,不论是选学 VB、C 或其他程序设计课程,程序设计的基本内容是一致的。掌握了前 4 个求解问题,即掌握了程序设计的基本内容。引导学生感受和领悟计算机分析问题和求解问题的过程、编程思想和基本方法,让学生一开始就在主观上明确程序设计课程的学习目标不仅仅是学会一门程序设计语言,更重要的是学会用计算机分析和解决问题的基本过程和思路,即学会如何把实际的问题转化为计算机可以解决的问题、如何用计算机的方法求解问题,从而在整个学习过程中,积极主动地注重计算思维的训练和培养。

3 通过程序设计中核心算法培养计算思维

掌握计算机求解问题的各种方法,是培养学生计算思维的关键。学生需掌握程序设计

的基本方法和基本问题的求解算法,深刻地理解计算机解决问题的思路和方法,逐步提升计算思维的能力。程序设计语言课程中有核心算法语句和基本算法。由于这些程序设计中的核心算法语句一般分散在各章节中,学到时才会向学生进行讲解,往往不会引起学生重视。因此,在讲授程序设计时(通常在第3章开始),可以先集中展示程序设计中的核心算法语句让学生作初步认识。

程序设计中的核心算法语句有:

- (1)计数器: $i = i + 1$ ($i = 0$);
- (2)累加器: $s = s + x$ ($s = 0$);
- (3)累积器: $p = p * x$ ($p = 1$);
- (4)累除器: $y = y / n$;
- (5)正负号变换器: $t = -t$ ($t = 1$ 或 $t = -1$);
- (6)终止标记: $x = -111$ ($x = 999$);
- (7)标记器: $f = 1/f = 0$ 或 $f = \text{true}/f = \text{false}$;
- (8)跟踪器: $p = i$;
- (9)变量交换: $t = a; a = b; b = t$;
- (10)变量值转移: $x = x1 + x2; x1 = x2; x2 = x$ 。

这样,学生就知道了这些特殊算法语句的含义及其在程序设计中所起的作用。这些算法语句不同于自然思维,是计算机的思维,程序设计始终离不开它们,要熟练地掌握和应用。

下面是应用这些核心算法语句解决典型问题(用VB程序语言编写)的几个教学示例。

教学示例一

求解问题: $s = 1 - 5 + 9 - \dots + (4n + 1)$ 。

```

dim n as integer
dim s as integer
dim t as integer
n = inputbox("n = ")
s = 0
t = -1
for i = 0 to n - 1 step 1
    t = -t
    s = s + t * (4 * i + 1)
next i
print "s = "; s
    
```

程序中应用了核心算法语句:累加器和正负号变换器。

教学示例二

求解问题:在1000到2000中找出能同时被37和91整除的自然数,如没有,请显示“找不到!”。

```

dim f as integer
f = 0
for i = 1000 to 2000 step 1
    if i mod 37 = 0 and i mod 91 = 0 then
        print i
        f = 1
    end if
next i
if f = 0 then print "找不到!"

```

程序中应用了核心算法语句:标记器。

教学示例三

求解问题:从键盘输入任意一批数,求它们的平均值。

```

dim x as integer, n as integer
dim s as single, v as single
n = 0
s = 0
x = inputbox("x = ")
do while x <> -111
    s = s + x
    n = n + 1
    x = inputbox("x = ")
loop
v = s/n
print "v = ";v

```

程序中应用了核心算法语句:累加器、计数器和终止标记。

4 计算思维能力培养的教学实践

在基于计算思维的理论背景下,我们在课堂理论教学和上机实践教学中始终主动、有意识地培养学生的计算思维能力,在教学实践中取得了良好效果。

在课堂理论教学过程中,我们尝试了几方面的教学改革。

(1)从计算思维的角度出发,将程序设计中的基本典型问题,按问题求解的要求进行步骤化。问题求解的一般步骤为:①建立数据模型(定义变量);②寻找解决方案(设计算法);③编程调试实践(有效优化);④完善解决问题(实现通用)。将实际问题的求解提升到计算机的思考即计算思维的高度,使学生在求解具体问题的过程中,逐步加深对计算思维本质的理解。

(2)将程序设计的语言与问题求解的过程分开。在讲解具体案例时,先提出具体问题,然后引导学生去体会为了解决问题而产生的大脑思考过程:已知数据是什么、数据类型如何表示、要求的结果是什么、求解方法如何实现等,让学生通过算法来理解计算机求解问题的思路,算法的描述可以用自然语言、形式代码或流程图等表示。学生理解了算法的基本思想后,再引入程序设计语言来编写代码并调试执行,实现问题的解决。

(3)程序设计中基本典型问题求解的几个教学示例(用 VB 程序语言编写),具体如下。

教学示例一:求最大值和最小值的基本典型问题

求解问题 1:输入 30 位学生的某门课程考试成绩,求出其中的最高分和最低分。

```

dim x as integer
dim max as integer, min as integer
max = -1
min = 101
for i = 1 to 30
    x = inputbox("x = ")
    if max < x then max = x
    if min > x then min = x
next i
print max, min
    
```

要从具体的成绩中(范围为 0~100 分)求出最大值和最小值, max 和 min 变量的初始值可以设定为: max=-1 和 min=101,这是以自然的思维来理解。

求解问题 2:输入 30 个整数,求出其中的最大值和最小值。

```

dim x as integer
dim max as integer, min as integer
x = inputbox("x = ")
max = x
min = x
for i = 2 to 30
    x = inputbox("x = ")
    if max < x then max = x
    if min > x then min = x
next i
print max, min
    
```

由于输入的是整数范围的数,要从一个较大的整数范围里正确求出最大值和最小值, max 和 min 变量的初始值可以用输入的第一个数作为 max 和 min 的初始值,这就是计算机的编程思想,即求解最大值和最小值的算法思想,对应的思维发生改变。

求解问题 3:输入任意一批数,求出其中的最大值和最小值。

```

dim x as integer
    
```

```

dim max as integer, min as integer
x = inputbox("x = ")
max = x
min = x
do while x<>-1
    if max<x then max = x
    if min>x then min = x
    x = inputbox("x = ")
loop
print max,min

```

对求解最大值和最小值的问题作进一步完善,不论输入多少个数据,都能正确实现求解,采用条件循环(do while/loop)结构解决了一般的通用性问题。对应的思维又发生改变。

教学示例二:实现平面图案输出的基本典型问题

求解问题 1:输出如图 1 所示的矩形图案。

```

for i = 1 to 10
    for j = 1 to 20
        print " * ";
    next j
    print
next i

```

这是双重循环输出平面图案的基本应用,外循环表示行,内循环表示列,程序实现了输出 10 行、每行 20 个星号的矩形图案。掌握了循环语句就能自然理解。

求解问题 2:输出如图 2 所示的三角形图案。

```

for i = 1 to 10
    for j = 1 to 2 * i - 1
        print " * ";
    next j
    print
next i

```

在图 1 所示矩形图案程序的基础上,将外循环变量结合到内循环中,设计适合的表达式,即可实现每行有变化的三角形图案。掌握内外循环的关系,思维理解更深入。

求解问题 3:输出如图 3 所示的等腰三角形图案。

```

for i = 1 to 10
    print tab(10 - i + 1);

```

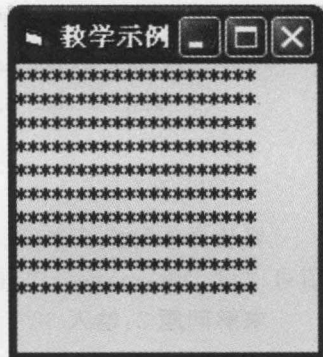


图 1

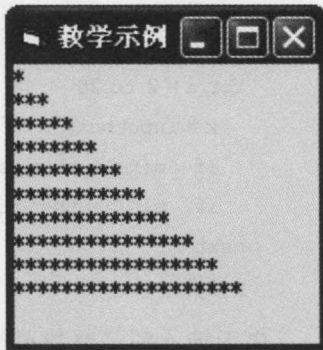


图 2

```

for j = 1 to 2 * i - 1
    print " * ";
next j
print
next i
    
```

在图 2 所示三角形图案程序的基础上,结合屏幕定位方式,对每行输出的位置进行改变,原来的三角形图案变成为等腰三角形图案。关键的定位语句实现了图案输出的大变化。

求解问题 4:输出如图 4 所示由数字组成的图案。

```

k = 0
for i = 1 to 10
    print tab(10 - i + 1);
    for j = 1 to 2 * i - 1
        k = k + 1
        if k > 9 then k = 0
        print k;
    next j
    print
next i
    
```

将图 3 所示由星号组成的图案,改为用数字符号实现,在原程序中加入计数器和条件语句,实现了图案的进一步变化。

求解问题 5:输出如图 5 所示由字母组成的图案。

```

k = 65
for i = 1 to 10
    print tab(10 - i + 1);
    for j = 1 to 2 * i - 1
        print chr(k);
        k = k + 1
        if k > 90 then k = 65
    next j
    print
next i
    
```

将图 4 所示由数字符号组成的图案,改为用字母符号实现,在程序中理解计数器的作用并结合 ASCII 码知识,稍作改变即可实现图案的变化。



图 3



图 4

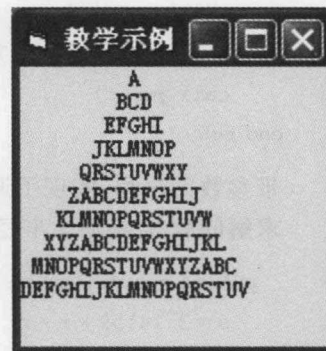


图 5

教学示例三:实现自定义过程函数的基本典型问题

求解问题 1:将输入半径求圆面积的问题用过程实现(不带参数)。

```
public sub ymj()
    dim r as single, s as single
    r = inputbox("r = ")
    s = 3.14159 * r * r
    form1.print "s=";s
end sub
private form_click()
    call ymj
end sub
```

一个过程即一个模块,不带参数的过程实质就是将源程序代码定义设计在一个模块中,使用时通过调用模块完成,从而理解自定义过程的意义和作用。

求解问题 2:将输入半径求圆面积的问题用过程实现(带参数)。

```
public sub ymj(byval r as single)
    dim s as single
    s = 3.14159 * r * r
    form1.print "s=";s
end sub
private form_click()
    dim r as single
    r = inputbox("r = ")
    call ymj(r)
end sub
```

带参数的过程,实现了调用模块时的数据传递作用,体现了模块的应用价值。

求解问题 3:将输入半径求圆面积的问题用过程实现(带参数并返回值)。

```
public sub ymj(byval r as single, byref s as single)
    s = 3.14159 * r * r
end sub
private form_click()
    dim r as single, s as single
    r = inputbox("r = ")
    call ymj(r,s)
    print "s=";s
end sub
```

过程定义时多增加一个参数(byref s as single),通过按地址传递的效果,一个真正有意义的过程模块便实现了。

求解问题 4:将输入半径求圆面积的问题用函数实现(带参数并返回值)。

```
public function ymj(byval r as single) as single
    s = 3.14159 * r * r
    ymj = s
end sub
private form_click()
    dim r as single, s as single
    r = inputbox("r = ")
    s = ymj(r)
    print "s = ";s
end sub
```

用 VB 程序语言中的函数模块设计同一个问题,过程和函数的区别就展现出来:如何自定义过程和函数、如何调用过程和函数、如何应用过程和函数等。通过上述输入半径求圆面积的简单问题的教学案例逐步展开,让学生的思维能力逐渐加强,从而掌握过程函数的知识内容。

上述几个教学示例均来自实际课堂教学,以计算机编程思想为导向,从一个简单求解问题出发,按正常的自然思维逐步过渡到计算机算法思想和计算思维上,从单一简单的程序设计逐步深入到综合复杂的程序设计,体现了掌握基本程序设计循序渐进的学习过程。

5 结 语

在程序设计课程中,通过程序设计方法和基本核心算法的传授来培养学生的计算思维是课程教学改革的核心。通过教学案例的精心设计和实践,可以证明,主动而有意识地将计算思维培养融入到理论教学和实践教学的各个环节,不仅有利于学生理解计算机求解问题的实现机制,更有利于学生用计算机学科的独特思维方式来思考问题和解决问题,对提高学生的思维能力、创造能力均有积极的作用,从而也提升了程序设计课程的教学效果。

参考文献

- [1] 唐培和,徐奕奕,王日凤. 计算思维导论[M]. 桂林:广西师范大学出版社,2012.
- [2] 陈国良,董荣胜. 计算思维与大学计算机基础教学[J]. 中国大学教学,2011(1):9—13.
- [3] 孙丽君,杨志强,高枚. 围绕计算思维培养的程序设计课程改革[J]. 计算机教育,2013(3):29—31.
- [4] 龚沛曾,杨志强. 大学计算机基础教学中的计算思维培养[J]. 中国大学教学,2012(5):53—56.
- [5] 李廉. 计算思维:概念与挑战[J]. 中国大学教学,2012(1):7—12.

农业信息化领域教学改革与实践

顾沈明 吴伟志 吴远红

浙江海洋学院,浙江舟山,316000

摘要:农业信息化领域是培养农业推广硕士研究生的重要领域之一。本文围绕农业信息化领域硕士研究生的培养目标与要求,根据自身的特点与教学实践,介绍农业推广硕士研究生的教学进程的内容及具体的改革方法与措施。

关键词:农业信息化;教学改革;教学进程

1 引言

近年来,为适应我国经济社会快速发展对各类专业人才的需求,国家出台了一系列促进专业学位研究生教育发展的政策、措施,大力促进专业学位研究生教育的发展^[1]。农业推广硕士专业学位经国务院学位委员会批准设置,定位于培养高层次应用型、复合型人才^[2]。农业推广硕士专业学位是为了满足我国建设社会主义新农村的宏伟目标和发展规划对高层次专门人才的迫切需求,增强科学技术对农业生产的支持度,提高科研成果转化为生产力的效率及加大面向农业的科技组织与服务能力而设定的新型学位类型^[3]。浙江海洋学院于2008年获得了农业推广硕士专业学位授予权,2009年获得了农业信息化领域学位招生资格。农业信息化领域专业硕士学位点自招生以来,我们边培养、边探索、边改革,在各级领导的大力支持下,克服重重困难,胜利完成培养任务。重点是以提高学生能力水平为出发点,组织教学内容,优化课程体系,开展教学改革与实践。

2 教学进程的调整

浙江海洋学院农业信息化领域硕士研究生培养方案主要是依据国家农业推广硕士教指委下发的《关于制订农业推广硕士专业学位研究生培养方案的指导意见》(农推指委〔2005〕5号),结合领域自身特点而制定的^[4]。学制两年,第一年主要学习课程有公共课、领域主干课和选修课等;第二年主要是实习、毕业论文及答辩等。另外,第一学期导师安排一些必修课程,第二学期导师安排做科研,第三学期准备发表小论文,第四学期准备毕业有关材料。

对于这样的安排,导师们普遍反映时间紧,培养效果不好。第一年学生忙于课程学习,第二年实习之后忙于毕业,对研究工作投入精力太少,研究不深入,水平很难提高。学校听取多方面的意见,结合多方面的因素,决定课程教学的时间调整为一个半学期。主要的课程学习集中在第一学期,留少量课程安排在第二学期的前半个学期。课程学习之后,除了

顾沈明 E-mail: gsm@zjou.edu.cn