Python程序语言应用在化工专业教学中的应用

黄河　张迎霜　亚力昆江·吐尔逊　黄雪莉　刘娜[通讯作者]
新疆大学化工学院/省部共建碳基能源资源化学与利用国家重点实验室　新疆 乌鲁木齐　830017

摘　要　Python是最热门的开源程序语言，受到越来越多的专业关注。为进一步培养学生运用编程语言解决化工专业问题，依托Python程序语言课程开展相关课程设计。学生将Python应用到物理化学、化工原理和反应工程等课程上，并开发了相应的应用小程序。通过课程设计提高学生的学习热情和用编程手段解决化工专业问题的能力。

关键词　Python；编程能力；课程设计；教学

Application of Python Programming Language in Chemical Engineering Education

Huang He, Zhang Ying-shuang, Yalikunjiang·Tuersun, Huang Xue-li, Liu Na[Corresponding author]
College of Chemical Engineering, Xinjiang University / State Key Laboratory of Carbon-Based Energy Resources Chemical and Utilization (Co-constructed by the Province and the Ministry), Urumqi 830017, Xinjiang Province, China

Abstract　Python, the most popular open-source programming language, has garnered increasing professional attention. To further cultivate students' ability to apply programming languages to solve problems in the field of chemical engineering, a relevant course design was implemented based on the Python programming language curriculum. Students applied Python to courses such as Physical Chemistry, Principles of Chemical Engineering, and Reaction Engineering, and developed corresponding small-scale application programs. This course design effectively enhanced students' learning enthusiasm and their capability to address specialized chemical engineering problems through programming.

Key words　Python; Programming Ability; Curriculum Design; Teaching

Python是一种完全开源的编程语言，其协议限制较少，版权由非营利组织Python软件基金会持有。作为一种高级编程语言，Python的语法接近自然语言，且对语法的要求相对较低。与C语言相比，Python代码更加简洁，编程量仅为C语言的20%。Python采用源代码执行的方式，能够快速实现代码的维护和跨操作系统的设计。作为一种轻量级的脚本语言，Python无须事先定义变量即可直接使用，并且对中文具有良好的兼容性[1]。

化工行业从业人需要具备通过编程语言解决化工专业计算问题的能力，将化工行业专业问题转化为计算机语言并解决。在化学工程与工艺专业学生教学过程中，以Python作为教学语言，并结合实际课程，将Python应用到分析化学、物理化学、化工热力学和反应工程等核心课程，一方面提高学生编程能力，另一方面提高学生将专业问题转化为计算机语言的能力。

1　化工专业课程特点

计算机语言作为化学工程与工艺专业的工程基础类课程，为支撑工程教育专业认证协会规定的工程知识体系和应用现代工具能力提供重要保障[2-3]。这要求毕业生能够针对化工及相关行业研究对象建立数学模型，并基于相关数据和条件利用先进方法或计算工具求解；能够选择、使用恰当的现代仪器、信息资源、工程工具和化工专业模拟软件，对化工及相关领域复杂工程问题进行分析、计算与设计。化工专业学生作为非计算机专业学生，重点关注程序语言在实际中的应用。

2　Python程序语言在化工专业核心课程中的应用

在课程设计过程中，学生运用Python解决分析化学、物理化学、反应工程、热力学及文献检索等课程中的问题。这些题目可以分为3类：第1类是利用Python计算教材中的例题，约占60%，主要集中在物理化学和分析化学领域；第2类是利用Python开发应用程序以进行查询，约占30%，如查询溶度积、分子量和元素信息；第3类是利用Python进行实验数据的处理与拟合，约占10%，主要集中于实验数据的分析。本文将选取3个具有代表性的案例进行详细展示[4]。

2.1 物理化学中的应用

接触角是物理化学中的重要概念，应用十分广泛。接触角是衡量化合物表面润湿性能的重要指标。通过计算接触角能够判断材料亲水和疏水性能，广泛应用于污水处理、石油开采、煤炭洗选和喷洒农药等行业。Python语言编程通过26行代码实现接触角计算，产生35 MB应用小程序，不同电脑环境可以直接使用（无须编码）。

Python程序代码由3个部分组成，详见图1。第一部分是库的引用，该代码使用了2个第三方库和1个标准。第二部分是名为calculate的函数编码，该函数将根据用户输入的值计算接触角状态。该部分内容是将专业问题转化为计算机语言，通过if-elif-else语句判断不同结果。第三部分是创建一个名为root的tkinter窗口，并设置其标题为“接触角计算”，详见图2。在窗口中使用grid布局方法添加标签、输入框和按钮，使用mainloop方法启动tkinter事件循环。该部分是计算成果显示部分，需要学生掌握计算过程的输入条件和输出结果。

第一部分

```
import tkinter as tk
from tkinter import messagebox
import math
```

第二部分

```
def calculate():
    try:
        r1 = float(entry_r1.get())
        r2 = float(entry_r2.get())
        r3 = float(entry_r3.get())
        cos_value = (r1 - r3) / r2
        if cos_value > 0:
            result = "浸湿"
        elif cos_value == 0:
            result = "铺展"
        else:
            result = "沾湿"
        messagebox.showinfo("结果", result)
    except ValueError:
        messagebox.showerror("输入错误", "请输入有效的数值")
```

第三部分

```
root = tk.Tk()
root.title("接触角计算")
tk.Label(root, text="请输入固体物质的表面张力（单位：牛每立方米）：").grid(row=0,
column=0, padx=10, pady=5)
entry_r1 = tk.Entry(root)
entry_r1.grid(row=0, column=1, padx=10, pady=5)
tk.Label(root, text="请输入液体物质的表面张力（单位：牛每立方米）：").grid(row=1,
column=0, padx=10, pady=5)
entry_r2 = tk.Entry(root)
entry_r2.grid(row=1, column=1, padx=10, pady=5)
tk.Label(root, text="请输入两种物质之间的界面张力（单位：牛每立方米）：").grid(row=2,
column=0, padx=10, pady=5)
entry_r3 = tk.Entry(root)
entry_r3.grid(row=2, column=1, padx=10, pady=5)
calculate_button = tk.Button(root, text="计算", command=calculate)
calculate_button.grid(row=3, columnspan=2, pady=10)
root.mainloop()
```

图1 Python程序代码



图2 小程序计算界面

2.2 反应工程中的应用

速率方程是反应工程的合成概念，不能直接获得，需要通过拟合实验数据获得。通过python可解决速率方程的确定问题，并对方程的偏差程度和拟合效果进行评价，能够极大程度降低数据工作量。利用Python语言开发了速率方程拟合小程序，程序代码详见图3。代码由5部分组成，第一部分是库引用，该代码引用了7个第三方库；第二部分创建一个tkinter窗口；第三部分构建一个open_file函数，用于打开保持数据的Excel表格；第四部分和第五部分是构建poltdata函数，第四部分读取Excel表格中的数据，第五部分对数据继续拟合和计算。该代码通过读实验数据表格，导入程序中自定义的函数进行图像拟合，并通过数学处理得出相关性系数R、皮尔逊系数p，同时用户输出反应拟合程度；将速率方程表达式在图像中进行数据标注并输出给用户，程序具备实时读取功能，读取结果列于图3。

第一部分

```
import matplotlib.pyplot as plt
from scipy.stats import linregress
from scipy.optimize import curve_fit
import numpy
import xlrd
from tkinter import *
import tkinter as tk
from tkinter import filedialog
```

第二部分

```
root = tk.Tk()
root.geometry('400x160')
label1=Label(root,text=" 文件表格形式：本程序只需提取表格的第一二行数据，
",fg="red",bg="yellow")
label1.pack()
label2=Label(root,text="默认第一行为反应物浓度，第二行为反应时间。",fg="red",bg="yellow")
label2.pack()
```

第二部分

```
def open_file():
    file_path = filedialog.askopenfilename()
    list = xlrd.open_workbook("{}".format(file_path))
    table = list.sheet_by_name('Sheet1')
    return table
```

第四部分

```
def plotdata():
    table = open_file()
    nrows = table.nrows
    ncols = table.ncols
    x=[float(table.cell_value(0, i)) for i in range(0, ncols)]
    y=[float(table.cell_value(1, i)) for i in range(0, ncols)]
    numpy.seterr(divide='ignore',invalid='ignore')#消除除数为 0 时候的错误#积分法
        def su(x, k, n):
          return k * numpy.power(x, n)
```

第五部分

```
    # 进行拟合
    popt, pcov = curve_fit (su, x, y)
    k=popt[0]
    n=popt[1]
    x1= numpy.arange(0, (1.1*max(x)), 0.1)
    y1 =k*x1**n
    fit_func = numpy.poly1d(popt)
    o=1-n
    slope, intercept, r_value, p_value, std_err = linregress(y**n, fit_func(x))
    plt.rcParams['font.sans-serif'] = ['SimHei']
    plt.xlabel('浓度 c   ')
    plt.ylabel('时间 t   ')
    plt.scatter(x, y,c='red',label='数据点')
    plt.plot(x1, y1,'--',label='拟合曲线')
    for i in range(len(x)):
        plt.annotate('(%s, %s)' % (str(x[i]), str(y[i])), xy=(x[i], y[i]), xytext=(1, -2),
textcoords='offset points',
                     ha='right', va='top', fontsize=8)
    plt.text(min(x)*1.1, max(y)*2,
             "速率方程:V={:.3f}*C**{:.3f}\n"
             "相关度系数 = {:.3f}   皮尔逊系数={:.3f}".format(k,o,r_value**2,p_value),
fontsize=12, ha="left", va="top")
    plt.legend()
    plt.title('浓度-时间曲线')
    plt.show()
btn = tk.Button(root, text="打开文件", command=plotdata)
btn.pack()
root.mainloop()
```

图3 Python程序代码

2.3 化工原理中的应用

道尔顿分压定律是化工原理中的重要概念，用于计算混合气体中各组分分压。利用Python开发35 MB小程序，通过输入气体组成和总压可实现对道尔顿分压求解。该代码由4部分组成，详见图4。第一部分是库引用，该代码引用了3个第三方库；第二部分用tkinter创建一个GUI界面，并添加相应的控件；第三部分构建了calculate函数，可以实现道尔顿分压的计算和绘制分压分布图；第四部分是计算函数和控件关联，当用户按下该按钮时调用计算程序，并更新结果。

```
import tkinter as tk
from tkinter import messagebox
import matplotlib.pyplot as plt
```

```
# 保存气体种类和按钮颜色信息
gas_info = [{"name": "绿", "color": "green", "num": 2}, {"name": "蓝", "color": "blue", "num": 3},
{"name": "黄", "color": "yellow", "num": 4},    {"name": "红", "color": "red", "num": 5}]
def create_gas_window(gas):
    gas_window = tk.Toplevel()
    gas_window.title(f"{gas['name']}气体输入")
    pressure_label = tk.Label(gas_window, text="恒定气压：")
    pressure_label.pack()
    pressure_entry = tk.Entry(gas_window)
    pressure_entry.pack()
    gas_entries = []
    for i in range(gas["num"]):
        label = tk.Label(gas_window, text=f"气体{i+1}的物质的量：")  # 修改标签
        label.pack()
        entry = tk.Entry(gas_window)
        entry.pack()
        gas_entries.append(entry)
```

```
    def calculate():
        gas_values = [float(entry.get()) for entry in gas_entries]
        total_moles = sum(gas_values)
        molar_fraction = [value / total_moles for value in gas_values]
        total_pressure = float(pressure_entry.get())  # 获取用户输入的恒定气压值
        partial_pressures = [molar_fraction[i] * total_pressure for i in range(len(molar_fraction))]
        messagebox.showinfo("计算结果", f"{len(gas_values)}种气体的分压力为：
{partial_pressures}")
        gas_window.destroy()
        # 创建条形统计图
        gases = ["气体" + str(i + 1) for i in range(gas["num"])]
        plt.bar(gases, partial_pressures)
        plt.xlabel("气体")
        plt.ylabel("分压力 / atm")
        plt.title(f"{gas['name']}气体分压力统计")
        plt.show()
    calculate_button = tk.Button(gas_window, text="计算", command=calculate)
    calculate_button.pack()
```

```
# 创建主窗口
window = tk.Tk()
window.title("气体种类选择")
def choose_gas(gas):
    create_gas_window(gas)
# 设置按钮之间的间距
button_padding = 10
# 计算按钮总宽度
total_button_width = sum(gas["num"] for gas in gas_info) * 100
# 将按钮的包装区域放置在主窗口的中间位置
window_width = total_button_width + (len(gas_info) + 1) * button_padding
window.geometry(f"{window_width}x300")
# 创建气体种类选择按钮
for gas in gas_info:
    button = tk.Button(window, text=gas["num"], bg=gas["color"], fg="black", command=lambda
g=gas: choose_gas(g))
    button.pack(side=tk.LEFT, padx=button_padding)
# 启动 GUI 事件循环
window.mainloop()
```

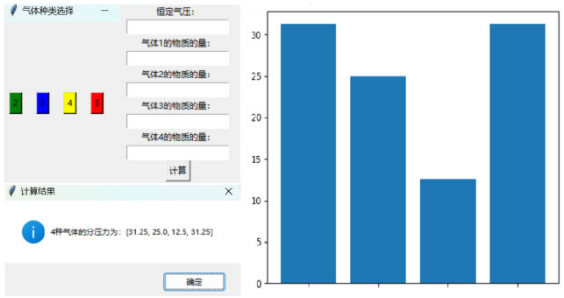

图4 Python代码和计算窗口

3 过程控制和预期效果

为保证教学效果，将课程设计分为两个阶段。阶段一，学生单独完成，题目选择限制在物理化学和分析化学例题；阶段二，学生以团队形式完成，每队5人且至少有1名少数民族同学，题目选择不做限制。阶段一学生100%能够完成，为阶段二工作开展提供基础。为保证设计进度，设置小组展示、小组自查和集中汇报环节。每周邀请1组在课堂展示进度和特点，每2周小组进行讨论并形成会议纪要，期末按照团队进行汇报。

课程设计实施后，期末考试＞90分的学生从10%提高到15%，学生对Python软件应用能力明显提高。以化工热力学课程为例，课后习题中使用编程进行迭代计算的，80%的学生选择Python语言，15%的学生选择MATLAB，15%的学生选择其他语言和手动计算[5]。

4 结束语

随着时代的进步，计算机在化工行业的应用日趋广泛。化工专业学生不仅要具有扎实的化工专业基础，还需要具备一定的计算机操作和运用能力。但是，化工专业属于非计算机专业，往往忽略了编程能力在化工行业中的应用。因此，在现有的化工专业课程体系中，强化学生用编程手段解决化工专业问题的能力，能够促进计算机技术和化工专业的结合，培养更符合时代要求的化工专业人才。

参考文献

[1]刘满兰,李建辉,关成斌.基于OBE理念的《Python程序设计语言》课程混合式教学方法探索[J].计算机工程与科学,2019,41(S1):203-206.

[2]中国工程教育专业认证协会.工程教育认证标准:T/CEEAA 00—2024[S].2024.

[3]嵩天,黄天羽,礼欣.Python语言:程序设计课程教学改革的理想选择[J].中国大学教学,2016,(2):42-47.

[4]杨庆春,李智伟,罗浩,等.计算机软件辅助化工设计课程教学的改革与实践[J].化工高等教育,2021,38(6):86-92.

[5]张青瑞,刘凯.计算软件在化学反应工程教学中的应用[J]化工高等教育,2019,36(4):90-95,105.

（本文编辑 曹 阳）