

实验题目：实验 2 tkinter 版小学数学口算题生成器设计与实现

实验环境：Python、PyCharm 等

一、实验目的

1. 熟悉 Python 标准库 tkinter 创建 GUI 应用程序的方法和步骤。
2. 熟练安装 Python 扩展库 python-docx。
3. 熟悉 Python 扩展库 python-docx 操作 Word 文档的方法。
4. 了解使用 Python 扩展库 python-docx 在 Word 文件中创建表格并写入数据的方法。
5. 了解小学生各年级数学知识的学习程度和口算题目要求。
6. 熟练使用 Python 标准库 random 中的函数。
7. 熟练使用 Python 标准库 os 中的函数。

二、实验内容

在小学一、二年级，只能口算 20 以内整数的加、减法；三、四年级可以口算超过 20 的整数四则运算；五年级以上可以口算带括号的式子。

编写程序，批量生成小学口算题，要求把生成的口算题写入 Word 文件中的表格。表格共 4 列，用户指定表格行数和题目适用年级。程序运行后界面如下图 1 所示。

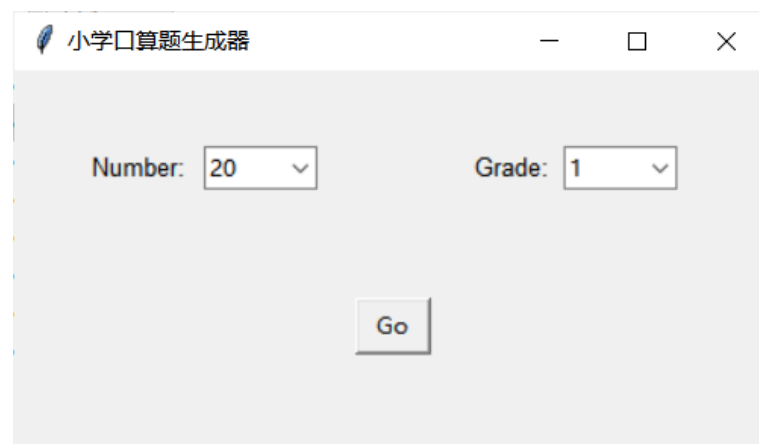


图 1 程序运行界面

三、实验步骤及结果

程序设计上，在对当地小学生各年级数学学习进度调查的之后，发现一至六年级的算数题目阶段划分十分明显，按照一至二、三至四、五至六分为 3 组较为笼统，例如三、四年级的口算水平使用同一套口算题目并不合适。为简化程序，将每一组的生成的口算题最高难度以较高的年级为准。

算法设计的源代码如下：

```
import random
import math
import tkinter as tk
from tkinter import ttk
import docx
from docx.shared import Pt
import os

Column = 4

def generate():
    """
    根据选中数据生成口算题
    """
    global number, grade
    Num = int(number.get())
    Gra = int(grade.get())

    # 创建文件
    Row = math.ceil(Num / Column)
    document = docx.Document()
    table = document.add_table(rows=Row, cols=Column)

    document.styles['Normal'].font.name = '宋体'
    document.styles['Normal'].font.size = Pt(12)

    # 用于随机生成不同年级水平的计算题目
    choi = [i for i in range(1, Gra + 1)]
    count = 0

    for row in range(Row):
```

```

for column in range(Column):
    level = random.choice(choi)
    # print(level)
    if level <= 2:
        equation = gene12()
    elif level <= 4:
        equation = gene34()
    elif level <= 6:
        equation = gene56()

    # 将算式写入表格中
    cell = table.cell(row, column)
    cell.text = equation
    count = count + 1
    # 生成的题目数量够了就退出
    if count == Num:
        break

# 保存文件, 文件名格式 Grade+年级+数量
filename = 'Grade' + str(Gra) + '-' + str(Num) + '.docx'
document.save(filename)
os.startfile(filename)

def gene12():
    """
    生成一、二年级的口算题, 20 以内的加减法, 不含负数
    :return:
    """
    # 生成题目
    scale = 20
    optrs = '+- '
    num1 = random.randint(1, scale)
    num2 = random.randint(1, scale)
    optr = random.choice(optrs)
    # 小学四年级才学负数, 本地的小学是四年级第一学期
    if optr == '-' and num1 < num2:
        num1, num2 = num2, num1
    equation = str(num1) + optr + str(num2) + '='
    return equation

def gene34():
    """
    生成三、四年级的口算题
    本地四年级学生最大可以计算两位数×三位数的乘除法, 没有学过分数; 四年级第一学期学

```

习负数

生成最大范围两位数×三位数的乘除法，保证可以整除

生成三个数的连续加减法，可能存在负数

```
"""
# 生成题目
scale1 = 999
scale2 = 99
optrs = '+-x÷'
choi = [1, 0] # 随机生成三个数的连续加减法或者乘除

num1 = random.randint(1, scale1)
num2 = random.randint(1, scale2)
optr = random.choice(optrs)
ways = random.choice(choi)

if ways == 1:
    # 生成乘除法
    if optr != '÷':
        equation = str(num1) + optr + str(num2) + '='
    else:
        # 四年级水平的除法应该保证可以整除
        if num1 < num2:
            num1, num2 = num2, num1
        while num1 % num2 != 0:
            num1 = random.randint(1, scale1)
            num2 = random.randint(1, scale2)
            if num1 < num2:
                num1, num2 = num2, num1
        equation = str(num1) + optr + str(num2) + '='
elif ways == 0:
    # 生成三个数的加减法
    optrs = '+- '
    num1 = random.randint(1, scale1)
    num2 = random.randint(1, scale1)
    num3 = random.randint(1, scale1)
    equation = str(num1) + random.choice(optrs) + str(num2) +
random.choice(optrs) + str(num3) + '='

return equation

def gene56():
    """
    生成五六年级的算数题目
    本地五年级学生学简单分数加减法，六年级可以计算复杂分数（使用通分等）
    """
```

```

生成包括+-×÷带括号混合的算术题，不考虑除法是否可以整除
"""

brackets = [1, 2] # 括号的位置，前两个数或后两个数
scale = 999

# 生成运算数
num1 = random.randint(1, random.randint(1, scale))
num2 = random.randint(1, random.randint(1, scale))
num3 = random.randint(1, random.randint(1, scale))
optrs = '+-×÷'
braPos = random.choice(brackets)

if braPos == 1:
    equation =
str(f"({num1}{random.choice(optrs)}{num2}){random.choice(optrs)}{num3}
)=")
    elif braPos == 2:
        equation =
str(f"{num1}{random.choice(optrs)}({num2}{random.choice(optrs)}{num3}
)=")
    return equation

if __name__ == '__main__':
    # 主窗口
    app = tk.Tk()
    app.title("小学口算题生成器")
    app.minsize(400, 200)
    app.maxsize(400, 200) # 控制窗口大小不可缩放

    # 题目数量
    numLabel = tk.Label(app, text='Number:', justify=tk.CENTER,
font=('Arial', 10))
    numLabel.place(x=40, y=40, width=50)

    number = tk.StringVar()
    numCombobox = ttk.Combobox(app, width=20, textvariable=number)
    numCombobox['values'] = (20, 50, 100, 150, 200, 250, 300)
    numCombobox.current(0)
    numCombobox.place(x=100, y=40, width=60)

    # 年级
    gradeLabel = tk.Label(app, text='Grade: ', justify=tk.CENTER,
font=('Arial', 10))

```

```

gradeLabel.place(x=240, y=40, width=50)

grade = tk.StringVar()
gradeCombobox = ttk.Combobox(app, width=20, textvariable=grade)
gradeCombobox['values'] = (1, 2, 3, 4, 5, 6)
gradeCombobox.current(0)
gradeCombobox.place(x=290, y=40, width=60)

# 确认按钮
buttonOK = tk.Button(app, text="Go", command=generate)
buttonOK.place(x=180, y=120, width=40)

app.mainloop()

```

程序算数题目时，高年级的算数题目中会包含较低年级的算数题目，例如六年级的算数题目会包含一至四年级的内容。一至五年级各生成若干道题目如下图 2 所示。

6+8=	9+3=	16-9=	9-6=
14+15=	13-11=	12-6=	17-15=
16-10=	13-10=	11+10=	11-5=
16+14=	19+10=	9+4=	17+7=
14-11=	7+1=	13-9=	6+10=
16+11=	2+10=	16-2=	16-6=
17+5=	16-5=	13+20=	12+14=
10+13=	20+10=	12-2=	12-9=
10+18=	18+19=	10+5=	8+2=
11+16=	19-2=	1+5=	18-1=
13-4=	11-10=	15-6=	18-9=
17-7=	17-13=	12-3=	15-12=
17-17=	11+16=		

图 2(a) 生成 50 道小学一年级题目

Number: 50

Grade: 2

样式

Go

15-7=	17-16=	13+16=	18-5=
18-8=	18-14=	17-4=	6-5=
19-16=	18-16=	18+11=	15-9=
3+15=	20-15=	16-1=	19-4=
17-12=	17+5=	17-13=	19-12=
18-6=	14-8=	14+8=	9-7=
8-7=	16+15=	9-1=	3+6=
10-2=	16-7=	19-11=	13-7=
17-13=	20-7=	16+19=	14-6=
5+1=	18+5=	9-7=	5+20=
8-5=	10+5=	1+1=	3+8=
8+19=	15-3=	11-7=	19-3=
19+3=	13+15=		

图 2(b) 生成 50 道小学二年级算数题目

Number: 20

Grade: 3

样式

Go

8+20=	20+14=	431+654+699=	3+7=
16+9=	274+25=	14-9=	684÷18=
18+2=	98×5=	7+4=	17-4=
519+77=	7+10=	385-39=	11-10=
678+711-603=	3+20=	2+11=	14-8=

图 2(c) 生成 20 道小学三年级题目

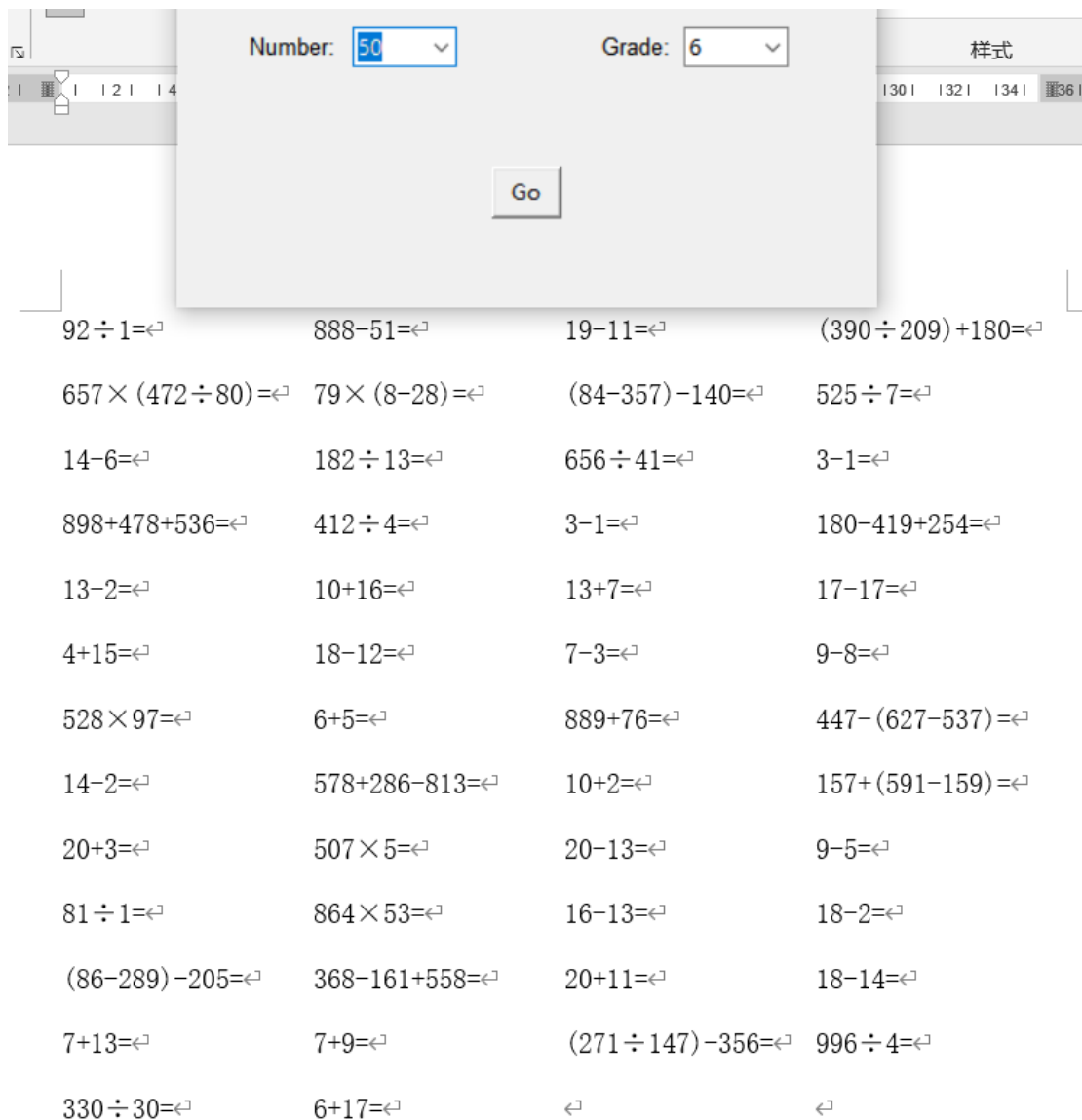


图 2(f) 生成 20 道小学六年级题目

四、 实验分析

1. 实验中需要针对各年级数学知识的学习程度设计算法生成算数题目，通过与当地某学校小学教师的交流大概了解了情况，合理设计算法实现实验要求。
2. 以课程上学到的关于 tkinter 创建 GUI 应用程序的步骤和方法为基础，结合网上查询资料，设计小学口算生成器的界面。对于相关知识的了解不再流于表面，在实践中对 GUI 应用程序的设计有了更深的掌握。
3. 在实验 1 的基础上，更多通过网上查询资料熟悉 Python 扩展库 python-docx 操作 Word 文档的方法，主要有创建 Word 文件、在 Word 文件中创建表格并写入数据。
4. 对标准库 random 和 os 中的一些函数掌握得更加熟练。