

期末考试试题 第二部分

第一题

1.函数 `find_name_value` 的功能描述如下：

这个函数将文件夹中的字符串和数值分离，转换成（字符串，数值）形式的元组。具体而言，`pattn` 规则将数值从文件名称中分离出来，并将文件名拆成一个列表 `rst`，这个列表的第一个元素是变量名（如 `'phi'`），第二个元素是数值字符串（如 `'0.1'`），如果有负数并且后面跟着 `n`，`rst` 列表的第三个元素会是 `'n'`，如果列表的第三个元素是 `n`，则会在数值前面加上一个 `-`。最后，这个函数将输出一个（字符串，数值）形式的元组

2.测试用例

- 正常输入
 - 测试1，输入 `phi0.1`，程序报错 `<unknown>:28: SyntaxWarning: invalid escape sequence '\d'`。经检查，是由于 `Python3` 将字符串文字解释为 `Unicode` 字符串，因此 `\d` 被视为转义的 `Unicode` 字符串。为避免报错，将 `pattern = '([-+]?\\d*\\.\\d+|[-+]?\\d+)'` 修改为 `pattern = '([-+]?\\d*\\.\\d+|[-+]?\\d+)'`。再次运行，输出 `('phi', 0.1)`，符合预期。
 - 测试2，输入 `xN14.2`，输出为 `('xN', 14.2)`，符合预期
 - 测试3，输入 `kappa0.5n`，输出为 `('kappa', -0.5)`，符合预期
- 异常输入
 - 测试4，输入 `kappa-0.5`，输出为 `('kappa', -0.5)`，符合预期
 - 测试5，输入 `kappa-0.5n`，程序报错，`ValueError: could not convert string to float: '-0.5'`，因为在判断是否包含负号后缀 `n` 时，代码将 `-` 号重复地添加到了数值部分，导致最终的字符串 `--0.5` 无法被转换为有效的浮点数。由此，增加了一道源代码的判断逻辑：

```
1 if float(rst[1]) < 0:
2     sign_str = ''
3 else:
4     sign_str = '-'
```

如果数值本身就带有 `-` 号，则 `sign_str` 为空值。再次测试，输入 `kappa-0.5n`，输出为 `('kappa', -0.5)`，符合预期

- 测试6，输入 `kappa`，输出为 `('kappa', None)`，符合预期
 - 测试7，输入 `kappa none`，输出为 `('kappa none', None)`，符合预期
 - 测试8，输入 `kappa 5`，输出为 `('kappa ', 5.0)`，符合预期
 - 测试9，输入 `n0`，输出为 `('n', 0.0)`，符合预期
- 边界输入
 - 测试10，输入 `phi0.1_xN14.2_kappa0.5n`，输出为 `('phi', 0.1)`，不符合预期。这种由多个 `<name><value>` 组合的文件名，需要新的模块来解决，添加代码如下：

```

1 def parse_file_name(file_name):
2     parts = file_name.split('_') # 按下划线拆分
3     results = []
4     for part in parts:
5         name, value = find_name_value(part) # 调用已有函数逐部分解析
6         results.append((name, value))
7     return results

```

通过迭代的方法，能够处理由_分割的多个期望值组合的文件名，此时的输出是一个嵌套元组的列表——`[('phi', 0.1), ('xN', 14.2), ('kappa', -0.5)]`

3.最终版本的代码如下：

```

1 import re
2
3 def find_name_value(folder_name):
4     '''Split the name of a data directory into a (name, value) tuple.
5
6     The format of ``folder_name``:
7
8         <name><value>
9
10        If the value is negative, it should be followed by a 'n'.
11
12        Examples:
13            ::
14
15                phi0.1          # should return 'phi', 0.1
16                xN14.2          # should return 'xN', 14.2
17                kappa0.5n       # should return 'kappa', -0.5
18
19        Args:
20            folder_name (str): the name of a :term:`data directory`.
21
22        Returns:
23            tuple: a tuple contains:
24
25            * name (str): variable name.
26            * value (float): value of the variable.
27        '''
28        pattern = '([-+]?\\d*\\.\\d+|[-+]?\\d+)'
29        rst = re.split(pattern, folder_name)
30        if len(rst) < 2:
31            return folder_name, None
32        name = rst[0]
33        valustr = rst[1]
34        sign_str = ''
35        if len(rst) > 2 and rst[2] == 'n':
36            if float(rst[1]) < 0:
37                sign_str = ''
38            else:
39                sign_str = '-'
40        value = sign_str + valustr

```

```

41         return name, float(value)
42     def parse_file_name(file_name):
43         parts = file_name.split('_') # 按下划线拆分
44         results = []
45         for part in parts:
46             name, value = find_name_value(part) # 调用已有函数逐部分解析
47             results.append((name, value))
48         return results
49     print(parse_file_name(file_name = "phi0.1_xN14.2_kappa0.5n"))
50     print(parse_file_name(file_name = "a1_b14n_n0_c0.2"))

```

运行以上代码，输入 `phi0.1_xN14.2_kappa0.5n`，输出为 `[('phi', 0.1), ('xN', 14.2), ('kappa', -0.5)]`；输入 `a1_b14n_n0_c0.2`，输出为 `[('a', 1.0), ('b', -14.0), ('n', 0.0), ('c', 0.2)]`。

第二题

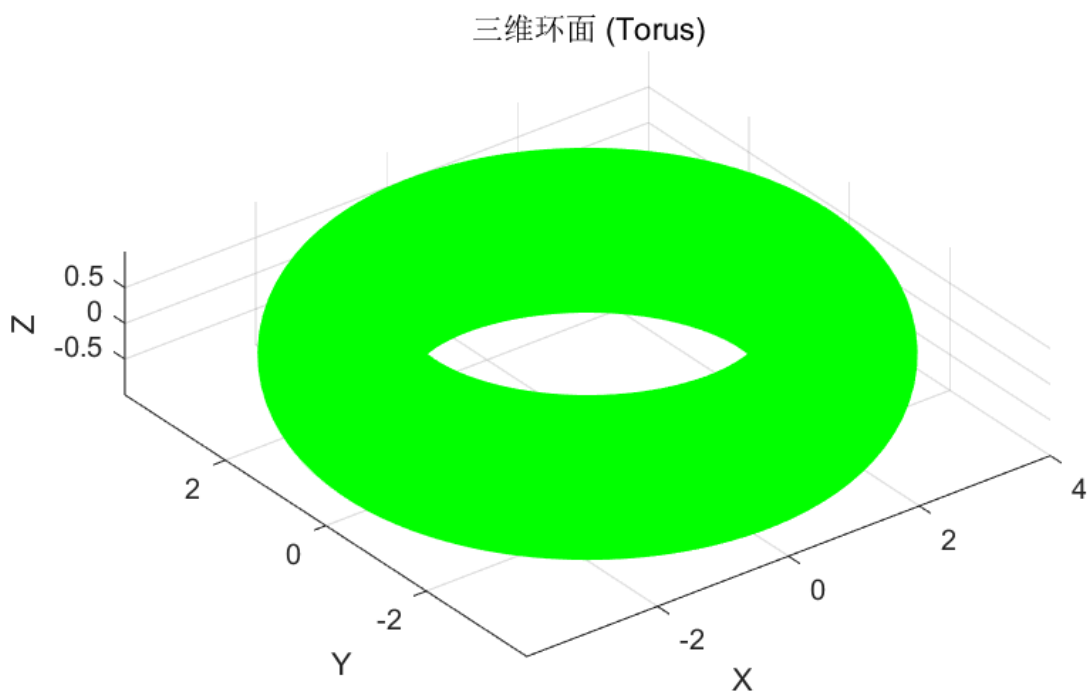
Matlab代码如下：

```

1  % 设置环面参数
2  R = 3; % 环面的大半径
3  r = 1; % 环面的小半径
4
5  % 创建参数网格，theta 和 phi 从 0 到 2*pi
6  theta = linspace(0, 2*pi, 100); % 纵向角度
7  phi = linspace(0, 2*pi, 100); % 横向角度
8
9  % 生成网格
10 [Theta, Phi] = meshgrid(theta, phi);
11
12 % 计算环面坐标
13 x = (R + r * cos(Theta)) .* cos(Phi);
14 y = (R + r * cos(Theta)) .* sin(Phi);
15 z = r * sin(Theta);
16
17 % 绘制三维环面
18 figure;
19 surf(x, y, z, 'FaceColor', 'g', 'EdgeColor', 'none'); % 设置颜色为绿色，去除边缘颜色
20
21 % 设置坐标轴，保证比例正确
22 axis equal;
23 xlabel('x');
24 ylabel('y');
25 zlabel('z');
26 title('三维环面 (Torus)');
27

```

结果图如下：



第三题

1.利用Mathematica计算以下无穷级数:

$$\sum_{n=1}^{\infty} \frac{1}{n^3 + n^2}$$

Mathematica语句如下: `Sum[1/(n^3 + n^2), {n, 1, Infinity}]`

输出为 $-1 + \pi^2/6$

2.利用Mathematica求如下定积分的值:

$$\int_0^{\infty} \frac{\sqrt{x} \ln(x)}{(x+1)^2} dx$$

Mathematica语句如下: `Integrate[(Sqrt[x] * Log[x]) / (x + 1)^2, {x, 0, Infinity}]`

输出为 π

第四题

Q: Find the solution of the following equation with respect to θ :

$$A \cos \theta + B \sin \theta + C = 0$$

A:

Let $x_1 = \cos \theta$ and $x_2 = \sin \theta$, then the solution is given by the intersection of the circle and the line:

$$\begin{aligned}x_1^2 + x_2^2 &= 1 \\ Ax_1 + Bx_2 + C &= 0\end{aligned}$$

We reformulate the equations in a parametric form:

$$\begin{aligned}|\boldsymbol{x}|^2 &= 1 \\ \boldsymbol{x}(t) &= \boldsymbol{a} + t\boldsymbol{b}\end{aligned}$$

where $\boldsymbol{x} = (x_1, x_2)$, $\boldsymbol{a} = (0, -C/B)$, $\boldsymbol{b} = (-C/A, C/B)$, and t is a parameter. The intersection points satisfy the following equation:

$$|\boldsymbol{a} + t\boldsymbol{b}|^2 = 1$$

which can be solved for t to find the intersection points:

$$t_{1,2} = \frac{-\boldsymbol{a} \cdot \boldsymbol{b} \pm \sqrt{(\boldsymbol{a} \cdot \boldsymbol{b})^2 - |\boldsymbol{b}|^2(|\boldsymbol{a}|^2 - 1)}}{|\boldsymbol{b}|^2}$$