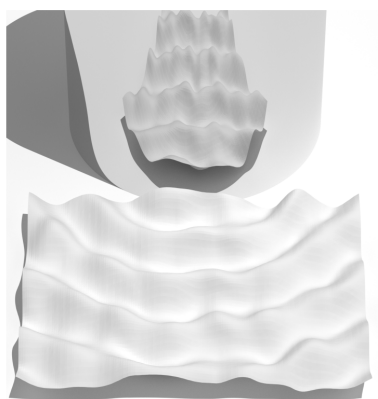


# 2024春-计算方法-第一次上机作业说明文档

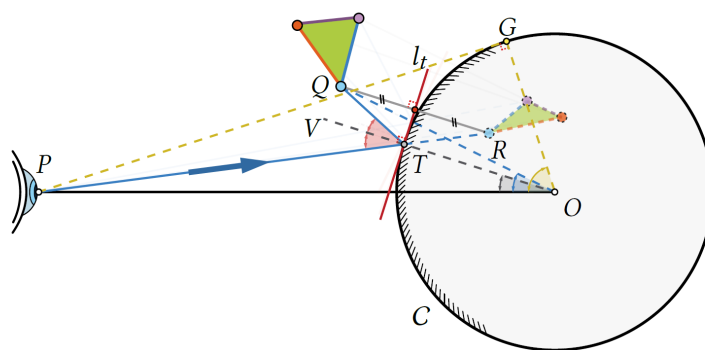
## 1 应用问题

### 1.1 背景

在物理光学模拟、计算机图形学等领域，一个常见的问题是求镜面反射中像的位置。对于平面镜面而言这是个简单的问题，但对于其他形状的镜面就不是了，比如圆柱形镜面（图1.a）。本次实验中，我们希望实验者编写计算机程序求解圆柱形镜面反射中像的位置的问题。



(a) 三维空间中通过圆柱形镜面看到物体的像



(b) 二维平面上的镜面反射示意图

图 1

### 1.2 问题

在三维空间中，给定观察者位置 $P$ ，圆柱形镜面的位置 $O$ 和半径 $r$ （假设高度无穷长），一点 $Q$ ，求 $Q$ 被观察到的像的位置 $R$ 。我们进一步注意到，由于镜面是圆柱形的，所以 $R$ 和 $Q$ 一定有相同的高度。因此我们将该问题转化到二维平面上（图1.b）。

如何解决该问题，由实验者自行思考并尝试，即可以通过解析的办法求解，也可以通过数值计算的方法求解。可参考文章《[Computational Mirror Cup and Saucer Art](#)》。

## 2 实验要求

1. 假设、输入和输出如下：

- 假设：在 $xy$ 平面上有一个镜面圆，可以反射光线，圆的圆心在原点处，半径为1；观察点在 $x$ 轴负半轴上；物点 $Q$ 在第二象限且在圆外。

- 输入:

观察点 $P$ ,  $P \in \{(x, y) | x < -1, y = 0\}$

物点 $Q$ ,  $Q \in \{(x, y) | x < 0, y > 0, x^2 + y^2 > 1\}$

- 输出: 反射点 $T$ 的位置, 像点 $R$ 的位置, 都用二维坐标 $(x, y)$ 表示 (如果你的算法的结果是其他表示方法, 请转换到二维坐标表示后输出)。

2. 算法要求: 实验者自行思考并设计算法。可参考引用的文章。

3. 程序要求:

- 强烈建议使用C、C++语言实现算法, 也能接受Python语言实现算法, 但不接受除上述语言之外的实现方式。
- 可以改变输入参数并测试, 输入参数包括三个:  $x_P, x_Q, y_Q$ 。比如, 使用main函数的参数argc、argv输入参数。又或者程序可以读取文件, 文件的每一行是一组输入参数。

4. 测试要求:

- 实验者自行测试如下数据, 精度达到4位有效数字相同或更高:

$P = (-2, 0), Q = (-1, 1)$ , 结果:  $T = (-0.885670, 0.464315), R = (-0.380057, 0.674993)$

$P = (-10, 0), Q = (-2, 1)$ , 结果:  $T = (-0.959312, 0.282350), R = (0.304214, 0.321811)$

- 实验者测试如下数据, 并在实验报告中给出结果:

$P = (-1.000001, 0), Q = (-2, 2)$ .

$P = (-2, 0), Q = (-1, 0.000001)$ .

$P = (-2.33, 0), Q = (-3, 1)$ .

$P = (-3, 0), Q = (-1, 0.5)$ .

$P = (-3, 0), Q = (-2, 10)$ .

$P = (-3, 0), Q = (-3, 1)$ .

$P = (-10, 0), Q = (-2, 1)$ .

$P = (-1024, 0), Q = (-8, 4)$ .

程序实现完毕后, 应撰写实验报告。实验报告中应包含如下内容:

1. 标题、学号、姓名。
2. 算法。如果你使用解析方法求解, 请列出推导公式。如果你使用数值方法求解, 请给出算法 (流程、伪代码或其他形式)。不论哪种方法, 都需要配以文字解释清楚明白。
3. 实验结果。输出必要的信息, 包含但不限于 $T$ 和 $R$ 。请截图附在报告中。
4. 请简要地以文字方式说明实验结果和分析, 并思考问题: 你的算法和程序中是否存在数值精度和稳健性问题, 如果存在该如何优化。

## 3 提交要求

### 3.1 提交方式

请提交源代码和实验报告。新建目录，并以“HW1-学号-姓名”方式命名，该目录下应包含如下内容：

- src\ （文件夹，存放你的源代码）
- report.pdf （你的实验报告）

将该文件夹以压缩包方式（压缩包名为“HW1-学号-姓名.zip”），发送到课程邮箱，comp\_method@163.com，邮件标题以同样方式命名。

请严格按照命名方式要求提交，不要交错邮箱，否则可能漏记成绩。

### 3.2 截止时间

在3月18日23:59分前提交。若有特殊情况请向助教说明。

## 4 参考

《Computational Mirror Cup and Saucer Art》