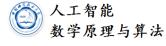


人工智能数学原理与算法

实验 4: 类和继承

2025年3月16日

目录



定义和使用类 二维平面上的点

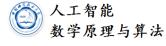
一元多项式

继承

近似计算一阶导数的公式 继承 覆盖

实验5: 类和继承

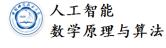
定义和使用类



面向对象的软件设计和开发技术是当前软件开发的主流技术,使得软件更易维护和复用。规模较大的软件大多是基于面向对象技术开发,以类作为基本组成单位。

在使用面向对象技术开发软件时,首先通过对软件需求的分析找到问题域中同一类的客观事物,称为对象。把对象共同的属性和运算封装在一起得到的程序单元就是类。类可作为一个独立单位进行开发和测试。以下举例说明。

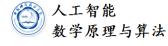
二维平面上的点



将二维平面上的点视为对象,抽象出其共同的属性和运算,定义一个类表示点。

- ▶ 点的属性包括×坐标、y坐标和名称(默认值为空串)。
- ▶ 点的运算包括:
 - ▶ 给定坐标创建一个点
 - ► 沿 x 轴平移
 - ► 沿y轴平移
 - ▶ 以另一个点为中心旋转
 - ▶ 计算与另一个点之间的距离等

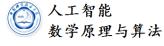
Point2D 类



程序 5.1 的第 2 行至第 34 行定义了一个 Point2D 类。

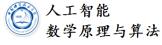
- ▶ 类的定义由 "class 类名 (父类)"开始。
- ▶ 第3行至第7行的方法___init___称为构造方法,用来初始 化新创建的对象的所有属性。
- ▶ 从一个类创建一个对象的语法是在类名后面加上一对圆括号,括号内可为空(表示无实参)或包含一些实参。如果实参的数量超过一个,它们之间用逗号分隔。这些实参必须和构造方法中除了 self 以外的那些形参在数量上相同,并且在顺序上一一对应。
- ▶ 定义在类内部的函数称为方法。

Point2D 类



- ► 在一个类中的所有方法中出现的属性都需要使用 "对象名."进行限定。
- ▶ self 是一个特殊的对象名,表示当前对象。
- ▶ 一个类中的所有方法的第一个形参都是 self。
- ▶ 对于一个对象调用其所属类的的方法的语法是在 "对象名. 方法名"后面加上一对圆括号,括号内可为空 (表示无实参)或包含一些实参。如果实参的数量超过一 个,它们之间用逗号分隔。这些实参必须和该方法中除了 self 以外的那些形参一一对应。
- ▶ Python 规定了类的一些特殊的方法,这些方法的名称都以 ____开始和结束。调用这些方法时可以使用简化语法。

一元多项式



程序 5.4 定义了一个类 Polynomial 表示一元多项式。

- ▶ 多项式中的每一项的指数和其对应的系数存储在一个字典 poly 中。如果在运算结果中某一项的系数的绝对值小于一 个预先定义的阈值 tol,则认为系数等于零,该项消失。
- ▶ 程序实现了多项式的加、乘、求值等运算。
- ▶ 程序定义了方法___call___。对多项式 p(x) 在 x=t 时求值的语法简化 p(t) 被转换为与其等价的方法调用 p.___call___(t)。
- ▶ 程序定义了方法___str___。该方法返回多项式的字符串表示,处理了多种特殊情形使得输出结果符合数学表达习惯。

近似计算一阶导数的公式



利用有限差分可以近似计算函数 f(x) 的一阶导数。以下按照 精确度从低到高次序列举公式依次称为一阶向前差分、一阶

 $f'(x) = \frac{f(x+h) - f(x)}{h} + O(h)$

$$f'(x) = \frac{f(x)}{x}$$

$$\frac{f(x) - f(x - h)}{h} + O(h)$$

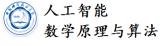
$$f'(x) = \frac{f(x) - f(x - h)}{h} + O(h)$$

$$f'(x) = \frac{f(x) - f(x - h)}{h} + O(h)$$

 $f'(x) = \frac{f(x + h) - f(x - h)}{2h} + O(h^2)$

$$\frac{f(x)-f(x-h)}{h}+O(h)$$

$$f'(x) = \frac{4}{3} \frac{f(x+h) - f(x-h)}{2h} - \frac{1}{3} \frac{f(x+2h) - f(x-2h)}{4h} + O(h^4)$$



如果每个公式都用一个类实现,则这些类都有属性f和h,并且它们的构造方法是相同的。对于每个公式,都需要比较其计算结果和精确结果的差别。这就导致了大量重复代码。

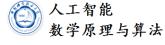
面向对象编程范式提供了继承机制,新类可从已有类获得属性和方法并进行扩展,实现了代码的重复利用。新类称为子类或派生类。已有类称为父类或基类。

可为这些公式类定义一个共同的父类 Differentiation,该类定义了初始化属性 f 和 h 的构造方法和比较结果的方法。通过继承,这些公式类可以获得这些属性和方法。 程序 5.5

子类可以对父类进行功能上的扩展,例如在子类中定义父类中没有的属性和方法。子类还可以重新定义从父类继承的方法,称为覆盖 (overriding)。覆盖的规则是子类中定义的某个方法和父类中的某个方法在名称、形参列表和返回类型上都相同,但方法体不同。

若一个子类定义了构造方法,则子类定义的构造方法覆盖了其父类的构造方法,此时若子类如果仍然需要从其父类继承属性,子类的构造方法必须包含语句 super().___init___();若一个子类未定义构造方法,则继承了其父类的构造方法,因此自动从其父类继承属性。程序 5.7

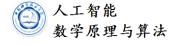
实验5: 函数和模块



本实验的目的是掌握以下内容:定义和调用函数,创建和使用模块。

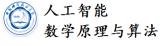
在 Blackboard 系统提交一个文本文件 (txt 后缀),文件中记录 每道题的源程序和运行结果。

1. 表示有理数的类

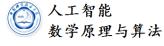


有理数的一般形式是 a/b, 其中 a 是整数, b 是正整数, 并且 当 a 非 0 时 |a| 和 b 的最大公约数是 1。实现 Rational 类表示 有理数和其运算。程序 5.8 已列出了部分代码, 需要实现标注 了"to be implemented"的函数。Rational 类的属性 nu 和 de 分别表示分子和分母。函数 add 、 sub 、 mul 和__truediv__分别进行加减乘除运算,然后返回一个新创建 的 Rational 对象作为运算结果。函数 eg 、 ne 、 __gt__、__lt__、__ge__和__le__比较两个有理数,返回 一个 bool 类型的值。这些函数对应的比较运算符分别是: == . != . > . < . >= . <= .

1. 表示有理数的类



例如表达式 Rational(6, -19) > Rational(14, -41) 在求值时被转换成方法调用 Rational(6, -19).___gt___(Rational(14, -41))。函数 test 测试这些函数。gcd 函数要求形参 a 和 b 都是正整数,如果其中出现 0 或负数,递归不会终止。

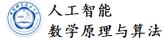


函数 f(x) 在区间 [a,b] 上的定积分可用区间内选取的 n+1 个点 x_i (i=0,1,...,n) (称为积分节点) 上的函数值的加权和近似计算:

$$\int_{a}^{b} f(x) dx \approx \sum_{i=0}^{n} w_{i} f(x_{i})$$

其中 w_i 是函数值 $f(x_i)$ 的权值,称为积分系数。不同的数值计算公式的区别体现在积分节点和积分系数上。

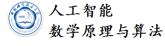
2. 定积分的数值计算



公式名称	积分节点的坐标和积分系数
复合梯形公式	$x_i = a + ih$ for $i = 0,, n$, $h = \frac{b-a}{n}$,
	$w_0 = w_n = \frac{h}{2}, w_i = h \text{for} i = 1,, n-1$
复合辛普森公式	$x_i = a + ih$ for $i = 0,, n$, $h = \frac{b-a}{n}$,
n必须是偶数	$ w_0 = w_n = \frac{h}{3}, w_i = \frac{2h}{3} \text{for} i = 2, 4,, n-2,$
若输入的 n 是奇数,则执行 $n = n + 1$	$w_i = \frac{4h}{3}$ for $i = 1, 3,, n - 1$
复合高斯-勒让德公式	$x_i = a + \frac{i+1}{2}h - \frac{\sqrt{3}}{6}h$ for $i = 0, 2,, n-1$,
n必须是奇数	$x_i = a + \frac{i}{2}h + \frac{\sqrt{3}}{6}h$ for $i = 1, 3,, n$,
若输入的 n 是偶数,则执行 $n = n + 1$	$h = \frac{2(b-\tilde{a})}{n+1}, w_i = \frac{h}{2}, \text{for} i = 0, 1,, n$

表: 定积分的几种数值计算公式

2. 定积分的数值计算



在程序 5.9 中实现 Integrator 类的 integrate 方法和它的三个子类,分别对应表中的三种公式。在每个子类中只需覆盖父类的 compute_points 方法计算并返回两个列表,它们分别存储了所有积分节点的坐标和积分系数。test() 函数用函数 $f(x) = (x\cos x + \sin x)e^{x\sin x}$ 和它的解析形式的积分函数 $F(x) = e^{x\sin x}$ 测试这三个公式的精确度。