Chapter 1

Differentiating Polynomials

If you had a function that gave you the height of an object, it would be handy to be able to figure out a function that gave you the velocity at which it was rising or falling. The process of converting the position function into a velocity function is known as *differentiation* or *finding the derivative*.

There are a bunch of rules for finding a derivative, but differentiating polynomials only requires three:

- The derivative of a sum is equal to the sum of the derivatives.
- The derivative of a constant is zero.
- The derivative of a nonconstant monomial at^b (a and b are constant numbers, t is time) is abt^{b-1}

So, for example, if I tell you that the height in meters of quadcopter at second t is given by $2t^3 - 5t^2 + 9t + 200$. You could tell me that its vertical velocity is $6t^2 - 10t + 9$

Exercise 1 Differentation of polynomials

Differentiate the following polynomials.

Working Space

Answer on Page 5

Notice that the degree of the derivative is one less than the degree of the original polynomial. (Unless, of course, the degree of the original is already zero.)

Now, if you know that a position is given by a polynomial, you can differentate it to find the object's velocity at any time.

The same trick works for acceleration: Let's say you know a function that gives an object's velocity. To find its acceleration at any time, you take the derivative of the velocity function.

Exercise 2 Differentation of polynomials in Python

Working Space

Write a function that returns the derivative of a polynomial in `poly.py`. It should look like this:

```
def derivative_of_polynomial(pn):
  ...Your code here...
```

When you test it in `test.py`, it should look like this:

```
# 3x**3 + 2x + 5
p1 = [5.0, 2.0, 0.0, 3.0]
d1 = poly.derivative_of_polynomial(p1)
# d1 should be 9x**2 + 2
print("Derivative of", poly.polynomial_to_string(p1),"is", poly.polynomial_to_string(d1))

# Check constant polynomials
p2 = [-9.0]
d2 = poly.derivative_of_polynomial(p2)
# d2 should be 0.0
print("Derivative of", poly.polynomial_to_string(p2),"is", poly.polynomial_to_string(d2))
```

Answer on Page 5

Appendix A

Answers to Exercises

Answer to Exercise 1 (on page 2)

Answer to Exercise 2 (on page 3)

```
def derivative_of_polynomial(pn):

    # What is the degree of the resulting polynomial?
    original_degree = len(pn) - 1
    if original_degree > 0:
        degree_of_derivative = original_degree - 1
    else:
        degree_of_derivative = 0

    # We can ignore the constant term (skip the first coefficient)
    current_degree = 1
```

```
    result = []

    # Differentiate each monomial
    while current_degree < len(pn):
        coefficient = pn[current_degree]
        result.append(coefficient * current_degree)
        current_degree = current_degree + 1

    # No terms? Make it the zero polynomial
    if len(result) == 0:
        result.append(0.0)

    return result
```

Index