6.101 Recitation 1: Week 0 Wednesday – Audio Processing Midpoint 2/7/24

R1 Participation Credit **Kerberos : _____@mit.edu**

Hand this sheet in at the end of recitation to get participation credit for today.

There are many ways to represent polynomials in Python. For today, we will represent polynomials in Python with lists, where the index represents the power and the element at that index represents the coefficient value. We will also assume all of our polynomials use the same single variable.

For example, the polynomial 3x^3 + x + 2 would be represented in Python as [2, 1, 0, 3]

Question 1: Given the code below, complete the test_poly_add function. Will the provided poly_add pass this test case? Why or why not? :

```
def poly_add(p1, p2):
    """
    Takes two polynomials as input and returns a
    new polynomial representing their sum. Does not modify inputs.
    """
    out = p1.copy()

    for i in range(len(p1)):

        out[i] += p2[i]

    return out



def test_poly_add():
    """
    Test case checking that poly_add can add
    x^2 + 3x  to  4x^3 + 7x - 8  and get 4x^3 + x^2 + 10x - 8
    """
```

Question 2: Complete the body of poly_mul below.

```
def poly_mul(p1, p2):
    """
    Takes two single variable polynomials as input and returns a
    new polynomial representing their product. Does not modify inputs.

    Ex: multiplying 6x^7 + 2x^4 + 5x^3 by 4x^2 + 3x  should result in the
      polynomial equivalent to 24x^9 + 18x^8 + 8x^6 + 26x^5 + 15x^4
    """
```

Question 3: Complete the body of poly_subtract below.

```
def poly_subtract(p1, p2):
    """
    Takes two single variable polynomials as input and returns a
    new polynomial representing their p1 - p2. Does not modify inputs.

    Ex: (4x^3 + 7x - 8) – (x^2 + 3x) should result in a polynomial equivalent
      To 4x^3 – x^2 + 4x - 8
    """
```