perplexity

solve this

I'll provide solutions for each of the Python programming tasks mentioned in the image:

1. Python program for arithmetic operations:

```
def arithmetic_operations(a, b):
    print(f"Addition: {a + b}")
    print(f"Subtraction: {a - b}")
    print(f"Multiplication: {a * b}")
    print(f"Division: {a / b}")
    print(f"Modulus: {a % b}")
    print(f"Exponentiation: {a ** b}")
    print(f"Floor division: {a // b}")

num1 = float(input("Enter first number: "))
num2 = float(input("Enter second number: "))
arithmetic_operations(num1, num2)
```

2. Python program to identify data types and mutability:

```
def identify_data_type(data):
    for item in data:
        data_type = type(item).__name__
        mutability = "mutable" if hasattr(item, '__setitem__') else "immutable"
        print(f"{item}: {data_type}, {mutability}")

mixed_data = [23, "hello", 3.14, (1, 2, 3), [5, 6, 7], {"name": "John"}, True]
identify_data_type(mixed_data)
```

3. Python program for string operations:

```
def string_operations(input_string):
    print(f"Uppercase: {input_string.upper()}")
    print(f"Lowercase: {input_string.lower()}")

    char = input("Enter a character to count: ")
    print(f"Occurrences of '{char}': {input_string.count(char)}")

    reversed_string = input_string[::-1]
    print(f"Reversed string: {reversed_string}")

user_input = input("Enter a string: ")
string_operations(user_input)
```

4. Python program for list operations:

```
natural_numbers = list(range(1, 21))

even_numbers = natural_numbers[1::2]
print(f"Even numbers: {even_numbers}")

reversed_list = natural_numbers[::-1]
print(f"Reversed list: {reversed_list}")

tuple_numbers = tuple(natural_numbers)
print(f"Tuple length: {len(tuple_numbers)}")
```

5. Python program for complex number operations:

```
def complex_operations(a, b):
    addition = a + b
    subtraction = a - b
    multiplication = a * b
    division = a / b

    print(f"Addition: Real = {addition.real}, Imaginary = {addition.imag}")
    print(f"Subtraction: Real = {subtraction.real}, Imaginary = {subtraction.imag}")
    print(f"Multiplication: Real = {multiplication.real}, Imaginary = {multiplication.ima
    print(f"Division: Real = {division.real}, Imaginary = {division.imag}")

num1 = complex(float(input("Enter real part of first number: ")),
               float(input("Enter imaginary part of first number: ")))
num2 = complex(float(input("Enter real part of second number: ")),
               float(input("Enter imaginary part of second number: ")))
complex_operations(num1, num2)
```

These Python programs address all the requirements specified in the image. Each program can be run independently to perform the requested operations.

⁂