In [ ]:
```python
import numpy as np
import pandas as pd
```

In [ ]:
```python
def init_ranges(data,cumulative):
    ranges = []
    for i in range(len(data)):
        symbol_range = []
        symbol_range.append(data[i][0])
        if i > 0:
            min_range = cumulative[i-1]
        else:
            min_range = 0

        max_range = cumulative[i]
        symbol_range.append(min_range)
        symbol_range.append(max_range)
        ranges.append(symbol_range)
    return ranges
```

In [ ]:
```python
def show_ranges(ranges):
    ranges_df = pd.DataFrame(ranges,columns=['Symbol','start range', 'end range'])
    print("------------ RANGES ------------")
    print(ranges_df)

```

In [ ]:
```python
def update_ranges(old_ranges,new_start, new_end,DECIMALS=5):
    #new val(C(s)) = new_start + C(s) * delta
    delta = float(new_end) - float(new_start)
    ranges = np.copy(old_ranges)
    for x in ranges:
        # symbol = x[0]
        old_min = np.round(float(x[1]),decimals= DECIMALS)
        old_max = np.round(float(x[2]),decimals= DECIMALS)

        new_min = float(new_start) + float(old_min) * delta
        new_min = np.round(new_min, decimals= DECIMALS)
        new_max = float(new_start) + float(old_max) * delta
        new_max = np.round(new_max, decimals= DECIMALS)

        x[1] = new_min
        x[2] = new_max

    return ranges
```

In [ ]:
```python
def encode_arithmetic(file, ranges, DECIMALS = 5):
    current_min_code = -1
    current_max_code = -1

    init_ranges = np.copy(ranges)
    current_ranges = init_ranges
    for char in file:
        for x in current_ranges:
            symbol = x[0]
            min_range = np.round(float(x[1]),decimals= DECIMALS)
            max_range = np.round(float(x[2]),decimals= DECIMALS)
            if symbol == char:
                print(f"symbol: {symbol} \nmin= {min_range}\nmax = {max_range}")
                current_min_code = min_range
                current_max_code = max_range
                current_ranges = update_ranges(init_ranges, min_range, max_range)
                show_ranges(current_ranges)
                print("==============================================")
    print("END")
    return current_min_code, current_max_code
```

In [ ]:
```
def arithmetic_to_binary(num, min_range, max_range):
    binary_string = "0."
    x = 0
    counter = -1
    while(True):
        num = num * 2
        int_part = int(num)
        num -= int_part
        x += int_part * 2**(counter)
        binary_string += str(int_part)
        if (x > min_range and x < max_range):
            break
        counter -= 1
        print(x)
    return binary_string
```

Usage

In [ ]:
```
# data = [(symbol, count), ... ()]

#data = [('A',100), ('B', 100), ('C',100), ('D',500), ('E',200), ('F',100), ('G',50
data = [('A',0.2), ('B', 0.3), ('C',0.25), ('D',0.25)]


#
```

In [ ]:
```
# data.sort(key=lambda a: a[1])
# print(data)
```

```
[('G', 50), ('H', 50), ('A', 100), ('B', 100), ('C', 100), ('F', 100), ('E', 200),
('D', 500)]
```

In [ ]:
```
freq = [count for symbol,count in data]
prob = freq / np.sum(freq)
cumulative = np.cumsum(prob)
#cumulative = [0.4, 0.7, 1]
```

In [ ]:
```
init_r = init_ranges(data,cumulative)
show_ranges(init_r)
```

```
------------ RANGES ------------
  Symbol  start range  end range
0      A         0.00       0.20
1      B         0.20       0.50
2      C         0.50       0.75
3      D         0.75       1.00
```

In [ ]:
```python
file = "ABD"
min_code, max_code = encode_arithmetic(file,init_r)
```

```
symbol: A
min= 0.0
max = 0.2
------------ RANGES ------------
  Symbol start range end range
0      A         0.0      0.04
1      B        0.04       0.1
2      C         0.1      0.15
3      D        0.15       0.2
==============================================
symbol: B
min= 0.04
max = 0.1
------------ RANGES ------------
  Symbol start range end range
0      A        0.04     0.052
1      B       0.052      0.07
2      C        0.07     0.085
3      D       0.085       0.1
==============================================
symbol: D
min= 0.085
max = 0.1
------------ RANGES ------------
  Symbol start range end range
0      A       0.085     0.088
1      B       0.088    0.0925
2      C      0.0925   0.09625
3      D     0.09625       0.1
==============================================
END
```

In [ ]:
```python
print(f"Arithmetic code for \"{file}\": \n\tminimum: {min_code}, \n\tmaximum: {max_
code = (min_code + max_code) / 2
print(f"Code (average) = {code}")
```

```
Arithmetic code for "ABD":
        minimum: 0.085,
        maximum: 0.1
Code (average) = 0.0925
```

converting Float to binary code

In [ ]:
```python
binary_code = arithmetic_to_binary(code, min_code, max_code)
print(f"binary code = {binary_code}")
```

```
0.0
0.0
0.0
0.0625
0.0625
0.078125
binary code = 0.0001011
```

Decoding

In [ ]:
```
binary_code = '0.01101'
```

In [ ]:
```
str_code = binary_code.split('.')[1]
list(str_code)
str_code
```

Out[ ]:
```
'01101'
```

In [ ]:
```
#works only on fractions
def bincode_to_decimal(code):
    str_code = binary_code.split('.')[1]
    number=0
    power = -1
    for digit in str_code:
        if int(digit) == 1:
            number += 2** power
        power -= 1
    return number
decimal_code = bincode_to_decimal(str_code)
print(f"number in decimals = {decimal_code}")
```

```
number in decimals = 0.40625
```

In [ ]:
```
def decode_arithmetic(code,ranges):
    init_ranges = np.copy(ranges)
    current_ranges = init_ranges

    decoded_string = ""
    MAX_ITER = 100
    COUNT = 0
    while True:
        for x in current_ranges:
            symbol = x[0]
            min_range = float(x[1])
            max_range = float(x[2])
            if  ((code >= min_range) and (code < max_range)):
                print(f"decoded_symbol: {symbol} \nmin= {min_range}\nmax = {max_ran
                decoded_string += symbol
                current_ranges = update_ranges(init_ranges, min_range, max_range)
                show_ranges(current_ranges)
                print("==============================================")
                if np.round(((min_range + max_range) / 2), decimals= 5) == np.round
                    print("END")
                    return decoded_string
            COUNT += 1
            if COUNT == MAX_ITER:
                return "Forced Exit"
```

In [ ]:

```
decoded_string = decode_arithmetic(decimal_code,init_r)
decoded_string
```

```
decoded_symbol: B
min= 0.2
max = 0.5
------------ RANGES ------------
  Symbol start range end range
0      A         0.2      0.26
1      B        0.26      0.35
2      C        0.35     0.425
3      D       0.425       0.5
==============================================
decoded_symbol: C
min= 0.35
max = 0.425
------------ RANGES ------------
  Symbol start range end range
0      A        0.35     0.365
1      B       0.365    0.3875
2      C      0.3875   0.40625
3      D     0.40625     0.425
==============================================
decoded_symbol: D
min= 0.40625
max = 0.425
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625      0.41
1      B        0.41   0.41562
2      C     0.41562   0.42031
3      D     0.42031     0.425
==============================================
decoded_symbol: A
min= 0.40625
max = 0.41
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625     0.407
1      B       0.407   0.40812
2      C     0.40812   0.40906
3      D     0.40906      0.41
==============================================
decoded_symbol: A
min= 0.40625
max = 0.407
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625    0.4064
1      B      0.4064   0.40662
2      C     0.40662   0.40681
3      D     0.40681     0.407
==============================================
decoded_symbol: A
min= 0.40625
max = 0.4064
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40628
```

```
1      B     0.40628   0.40632
2      C     0.40632   0.40636
3      D     0.40636    0.4064
==============================================
decoded_symbol: A
min= 0.40625
max = 0.40628
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40626
1      B     0.40626   0.40626
2      C     0.40626   0.40627
3      D     0.40627   0.40628
==============================================
decoded_symbol: A
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
```

```
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
```

```
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
```

```
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
decoded_symbol: B
min= 0.40625
max = 0.40626
------------ RANGES ------------
  Symbol start range end range
0      A     0.40625   0.40625
1      B     0.40625   0.40626
2      C     0.40626   0.40626
3      D     0.40626   0.40626
==============================================
```

```
Out[ ]:  'Forced Exit'
```

To DO

handling Forced Exit problem in Decoding