Name: Soumen

Batch: B1

Roll No:16010420133

ITC TUTORIAL 4B

Q1. Write a program in C or Python to implement the Arithmetic coding method.

The program should display the intermediate values of cumulative probability at each step (each successive symbol in the string) and the final encoding range for the following strings: -

i. “PEPPER”

ii. “SUCCESS”

iii. “PINGPONG”

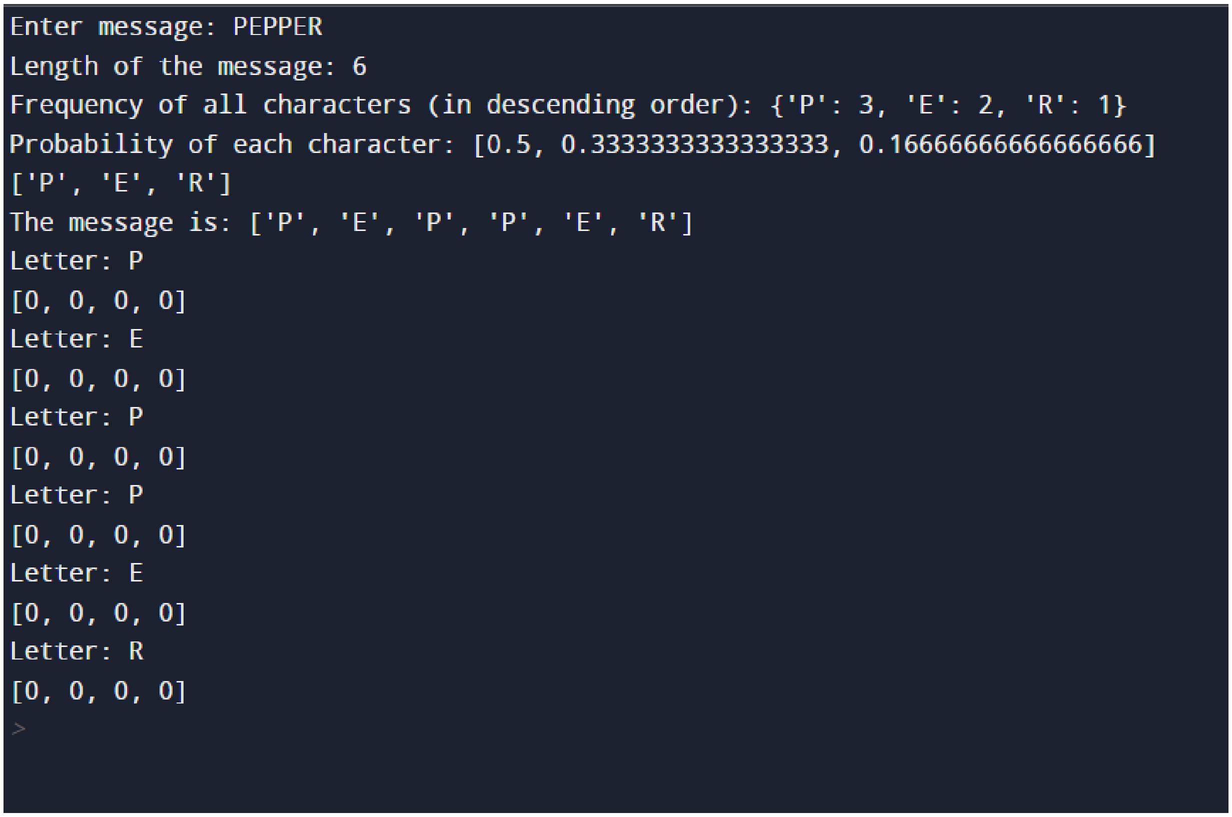
Program

```
message = list(input("Enter message: "))
for i in message:
    if i == ' ':
        message.remove(i)
length = len(message)
print("Length of the message:",length)
a = {}
b = []
for i in message:
    if i not in b:
        b.append(i)
        a[i] = message.count(i)
dict1 = {k: v for k, v in sorted(a.items(), key=lambda item: item[1], reverse=True)}
print("Frequency of all characters (in descending order):",dict1)
c = list(dict1.keys())
d = list(dict1.values())
```

```
p_arr = [i/length for i in d]

print("Probability of each character:",p_arr)

cp_arr = [0]

sum = 0

for i in p_arr:

    sum += i

    cp_arr.append(sum)

# print(cp_arr)

print(c)

print("The message is:",message)


for i in message:

    print("Letter:",i)

    position = int(c.index(i))

    range_now = cp_arr[position:position+2]

    a = range_now[0]

    b = range_now[1]

    summ = range_now[0]

    add = (b - a)/(len(c))

    new_arr = []

    for j in range(len(c)+1):

        new_arr.append(summ)

        summ += add

    print(new_arr)

    del range_now

    del cp_arr

    position = 0

    range_now = []
    cp_arr = new_arr.copy()
```

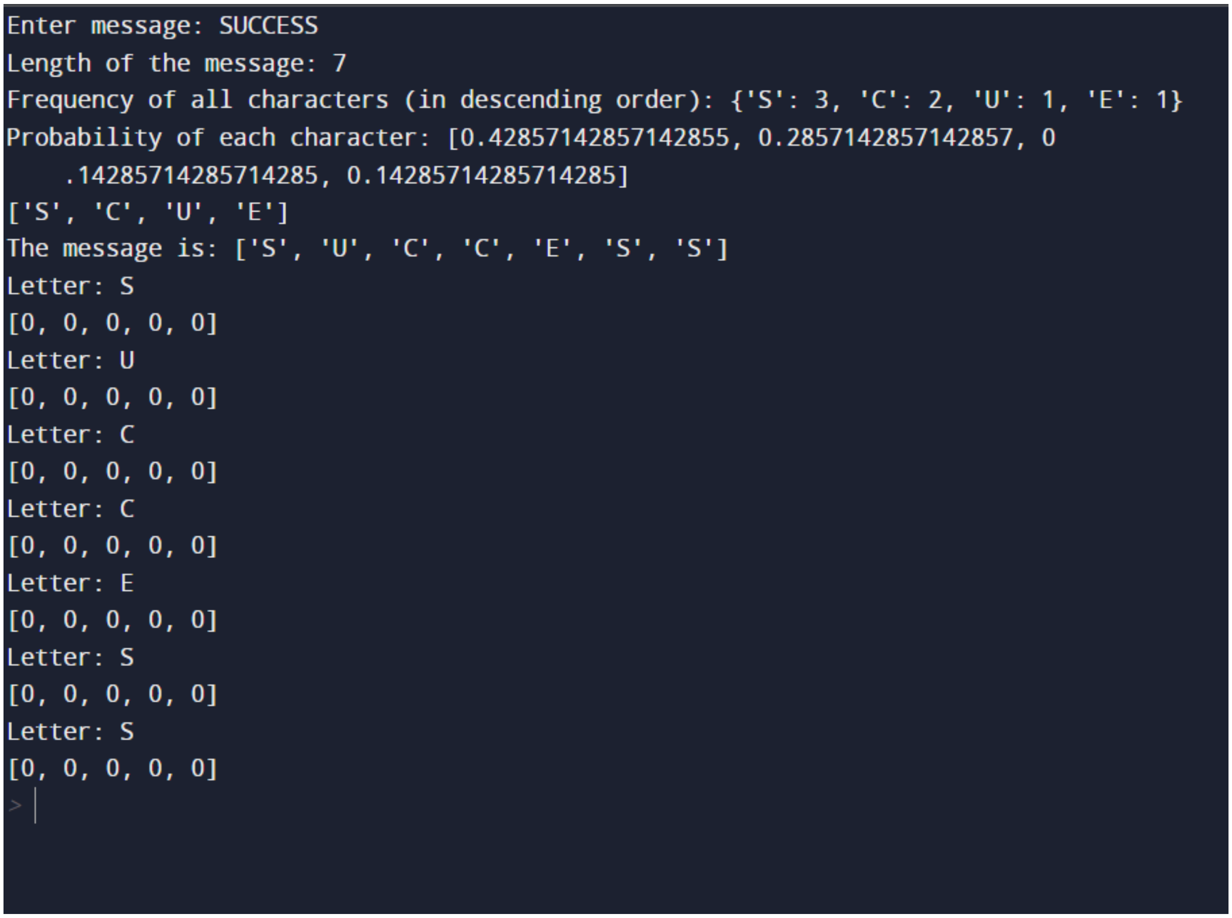
Output:

PEPPER



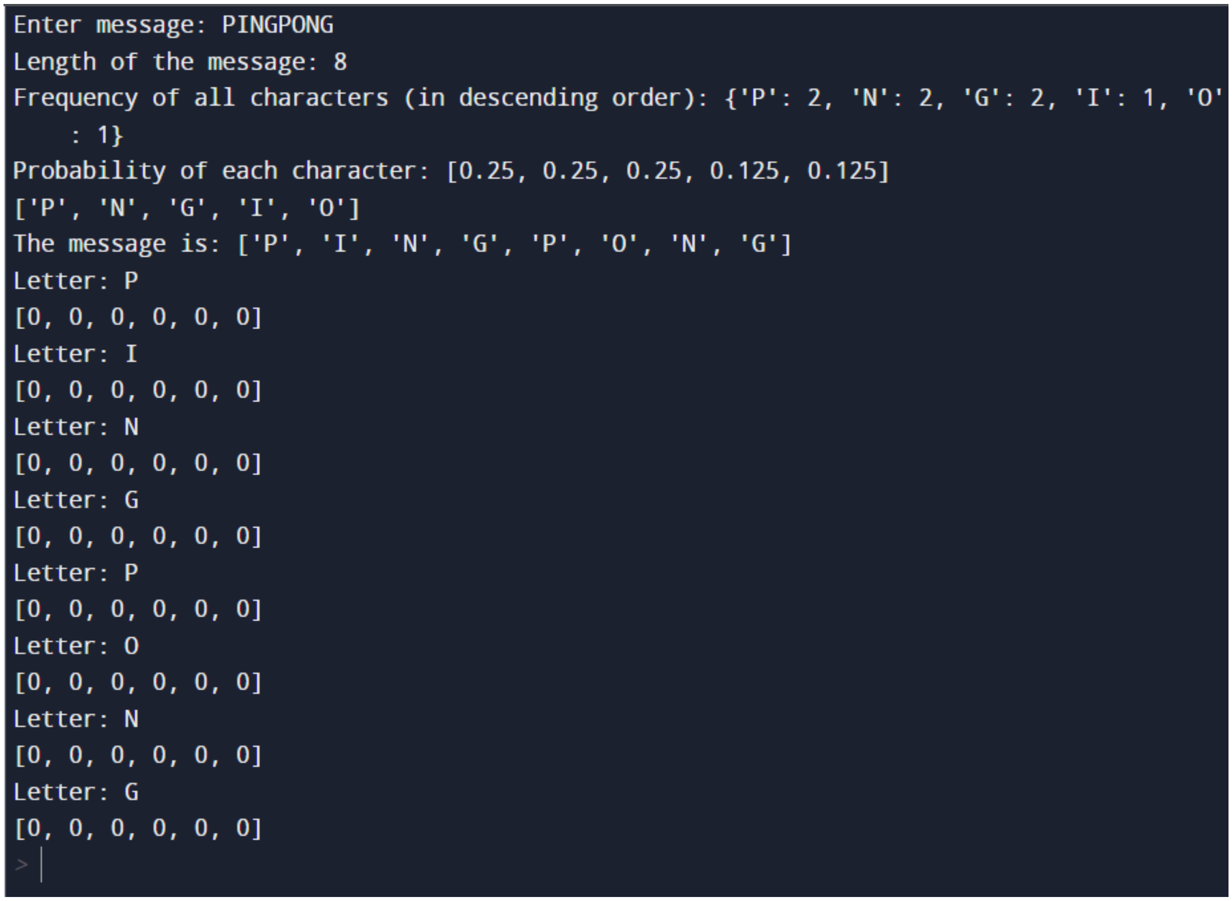
```
Enter message: PEPPER
Length of the message: 6
Frequency of all characters (in descending order): {'P': 3, 'E': 2, 'R': 1}
Probability of each character: [0.5, 0.3333333333333333, 0.16666666666666666]
['P', 'E', 'R']
The message is: ['P', 'E', 'P', 'P', 'E', 'R']
Letter: P
[0, 0, 0, 0]
Letter: E
[0, 0, 0, 0]
Letter: P
[0, 0, 0, 0]
Letter: P
[0, 0, 0, 0]
Letter: E
[0, 0, 0, 0]
Letter: R
[0, 0, 0, 0]
>
```

SUCCESS



```
Enter message: SUCCESS
Length of the message: 7
Frequency of all characters (in descending order): {'S': 3, 'C': 2, 'U': 1, 'E': 1}
Probability of each character: [0.42857142857142855, 0.2857142857142857, 0
    .14285714285714285, 0.14285714285714285]
['S', 'C', 'U', 'E']
The message is: ['S', 'U', 'C', 'C', 'E', 'S', 'S']
Letter: S
[0, 0, 0, 0, 0]
Letter: U
[0, 0, 0, 0, 0]
Letter: C
[0, 0, 0, 0, 0]
Letter: C
[0, 0, 0, 0, 0]
Letter: E
[0, 0, 0, 0, 0]
Letter: S
[0, 0, 0, 0, 0]
Letter: S
[0, 0, 0, 0, 0]
>
```

PINGPONG



```
Enter message: PINGPONG
Length of the message: 8
Frequency of all characters (in descending order): {'P': 2, 'N': 2, 'G': 2, 'I': 1, 'O'
    : 1}
Probability of each character: [0.25, 0.25, 0.25, 0.125, 0.125]
['P', 'N', 'G', 'I', 'O']
The message is: ['P', 'I', 'N', 'G', 'P', 'O', 'N', 'G']
Letter: P
[0, 0, 0, 0, 0, 0]
Letter: I
[0, 0, 0, 0, 0, 0]
Letter: N
[0, 0, 0, 0, 0, 0]
Letter: G
[0, 0, 0, 0, 0, 0]
Letter: P
[0, 0, 0, 0, 0, 0]
Letter: O
[0, 0, 0, 0, 0, 0]
Letter: N
[0, 0, 0, 0, 0, 0]
Letter: G
[0, 0, 0, 0, 0, 0]
>
```