Report
Compression and Decompression
IN IMAGE

Implement & Analyze

Research

COMPRESSION

Photo Gallery.

R, G, B of Photo

File code after compressing

DEcompression

File code after compressing

R, G, B of Photo

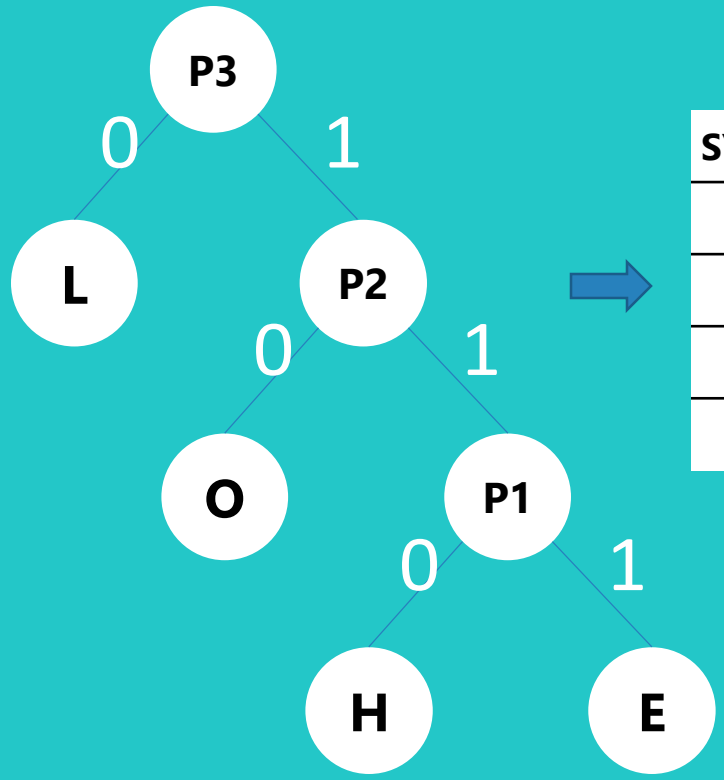
Photo Gallery.

Huffman Coding

COMPRESSION

HELLO

symbol	H	E	L	O
occurence	1	1	2	1

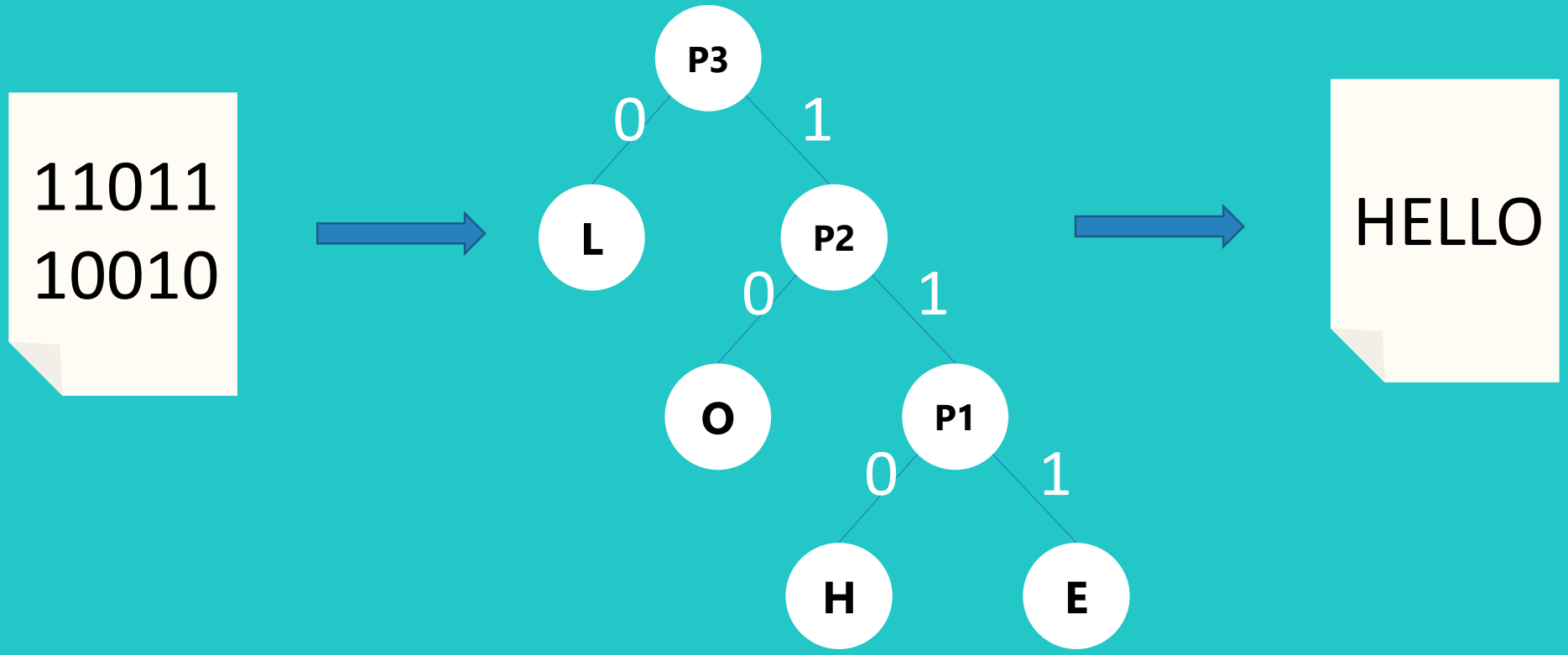


SYMBOL	CODE
L	0
O	10
H	110
E	111

11011
10010

Huffman Coding

DECOMPRESSION



SO WE NEED TO STORE THIS TREE!

LZW Coding

COMPRESSION

AAABB
BEDFE
D

Initial dictionary

CODE	STRING
1	A
2	B
3	D
4	E
5	F

CODE	STRING
1	A
2	B
3	D
4	E
5	F
6	AA
7	AB
8	BB
9	BE
10	ED
11	DF
12	FE

16128
24351
0

LZW Coding

DECOMPRESSION

Parallel

16128
24351
0

CODE	STRING
1	A
2	B
3	D
4	E
5	F

WE NEED TO STORE THE INITIAL DICTIONARY!

CODE	STRING
1	A
2	B
3	D
4	E
5	F
6	AA
7	AB
8	BB
9	BE
10	ED
11	DF
12	FE

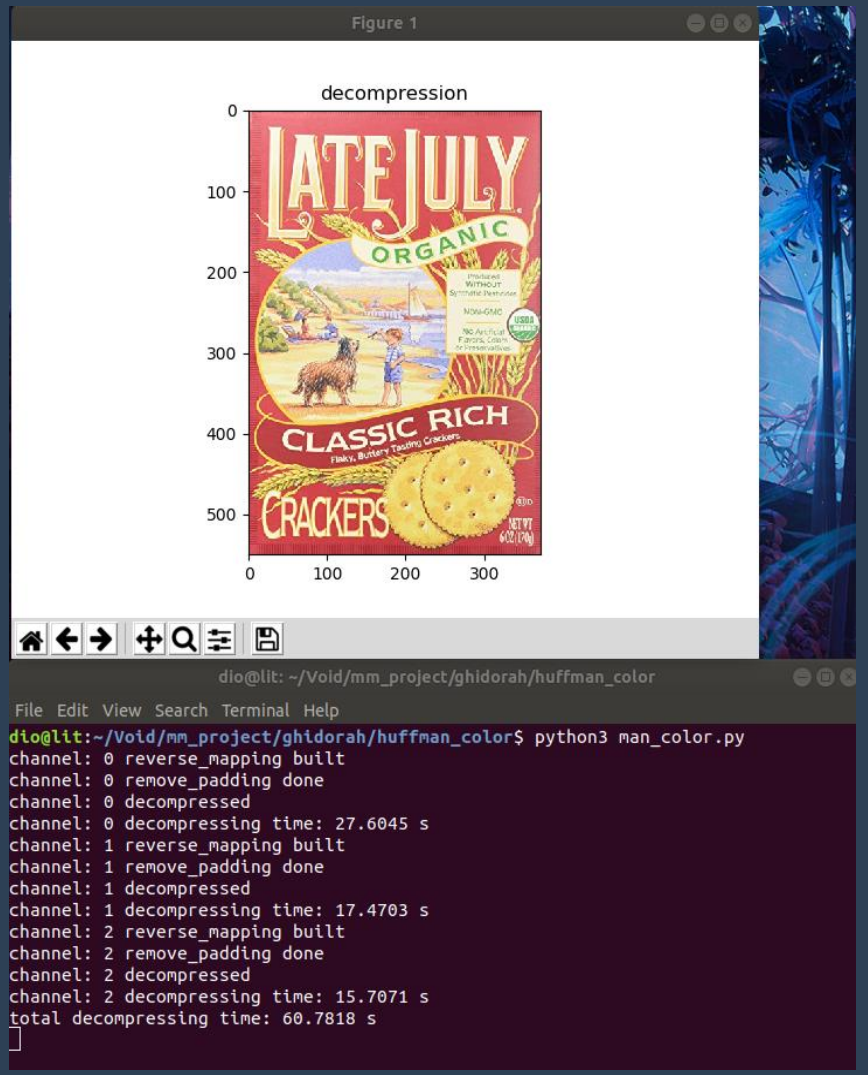
AAABB
BEDFE
D

IMPLEMENTATION AND RESULT

Original size: 615450 bytes

Total compression size: 555785 bytes

Total compressing time: 0.5559s



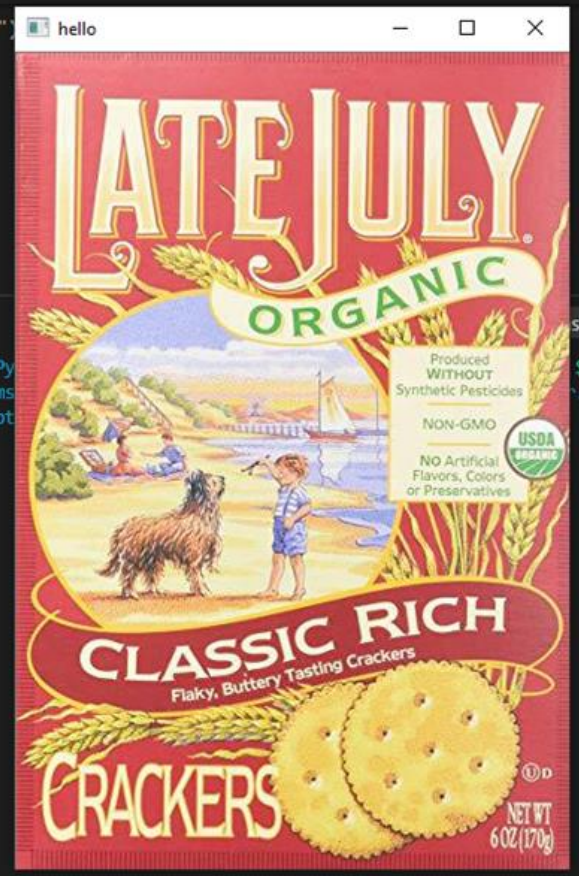
Total decompressing time: 60.7818s

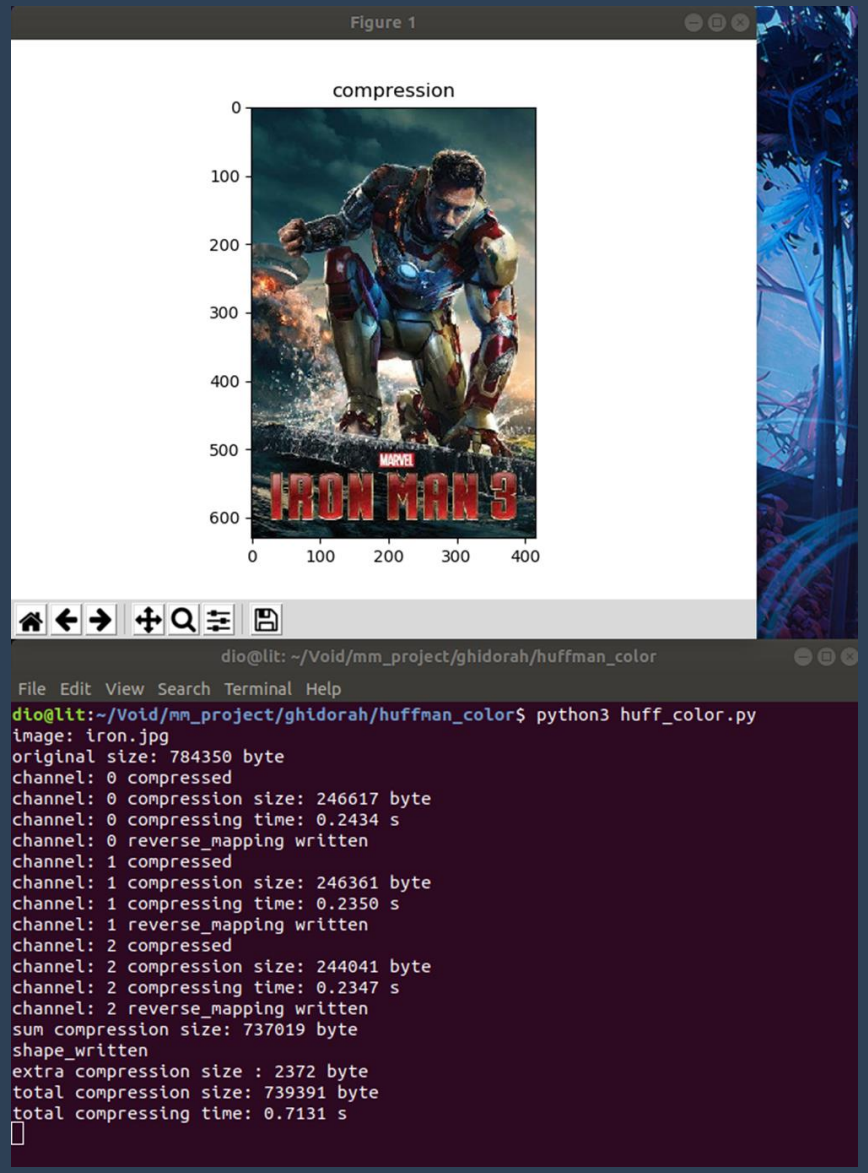
```
Image: hopbanh.jpg
channel: 0 keys written
channel: 0 values written
channel: 0 encoded
Done encode channel 0 : --- 1.51 seconds ---
channel: 1 keys written
channel: 1 values written
channel: 1 encoded
Done encode channel 1 : --- 1.46 seconds ---
channel: 2 keys written
channel: 2 values written
channel: 2 encoded
Done encode channel 2 : --- 1.82 seconds ---
shape written
Done encode: --- 4.79 seconds ---
```

```
70      img = np.reshape(output, (shape[0], shape[1])).astype("uint8")
71      return img
72
73
74  if __name__ == "__main__":
75      fCode, fKey, fValue, fShape = ReadFile()
76      code, dictionary, value = list(), list(), list()
77      shape = ReadFileToShapeImage(fShape)
78      total = 0
79      for i in range(3):
```

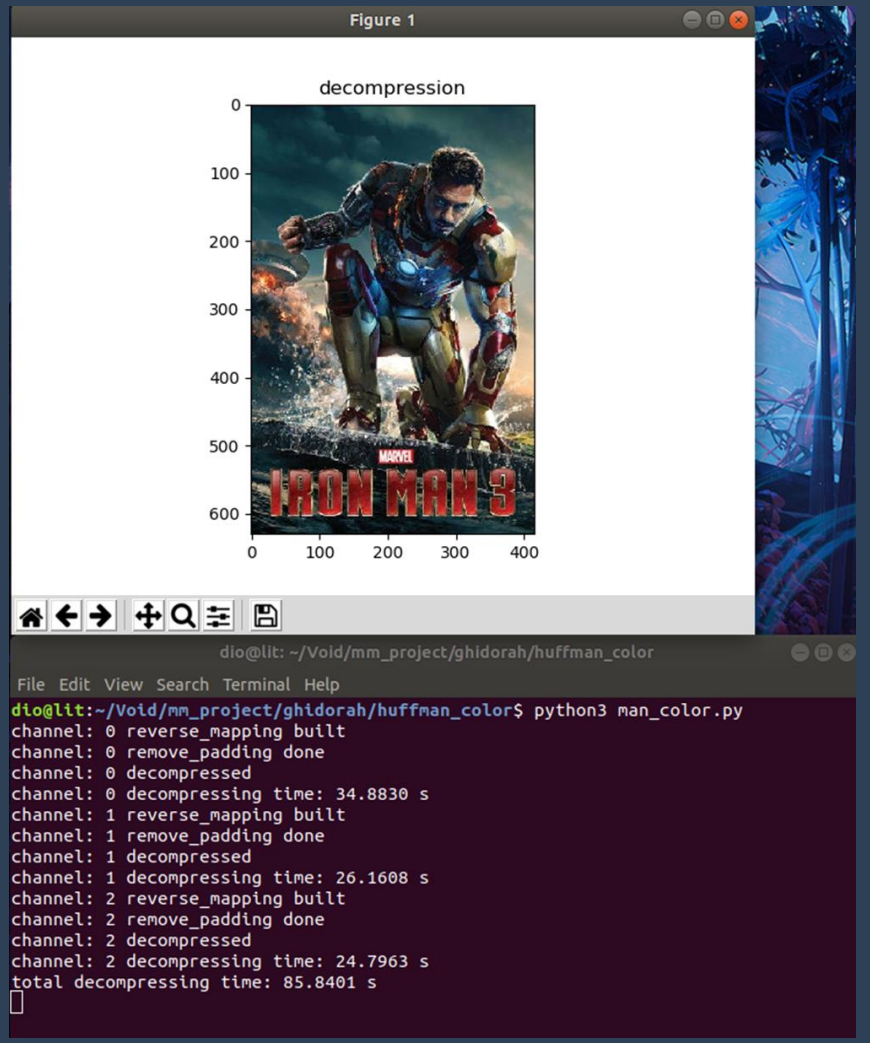
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\goku\Documents\Python Scripts\TTDPT> cd 'c:\Users\goku\Documents\Py
1'; & 'C:\Users\goku\Anaconda3\python.exe' 'c:\Users\goku\.vscode\extensions\ms
t' '--host' 'localhost' '--port' '51566' 'c:\Users\goku\Documents\Python Script
Done decode channel 0 : --- 3.56 seconds ---
Done decode channel 1 : --- 2.27 seconds ---
Done decode channel 2 : --- 1.86 seconds ---
Done decode: --- 7.69 seconds ---
```





Original size: 784350 bytes
Total compression size: 739391 bytes
Total compressing time: 0.7131s



Total decompressing time: 85.8401s

```
Image: ironman.jpg
channel: 0 keys written
channel: 0 values written
channel: 0 encoded
Done encode channel 0 : --- 2.2 seconds ---
channel: 1 keys written
channel: 1 values written
channel: 1 encoded
Done encode channel 1 : --- 2.51 seconds ---
channel: 2 keys written
channel: 2 values written
channel: 2 encoded
Done encode channel 2 : --- 2.19 seconds ---
shape written
Done encode: --- 6.9 seconds ---
```

encode.py | LZWDecode.py ×

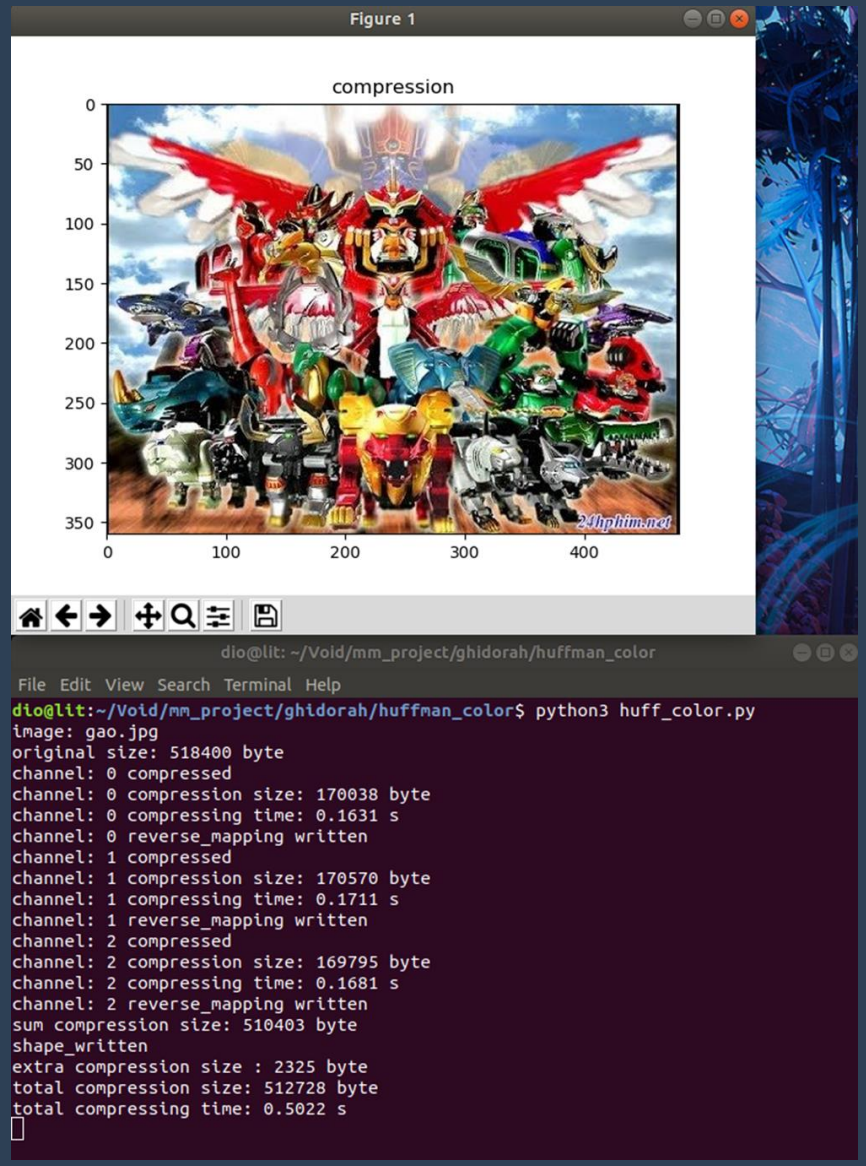
LZWDecode.py ▸ ...

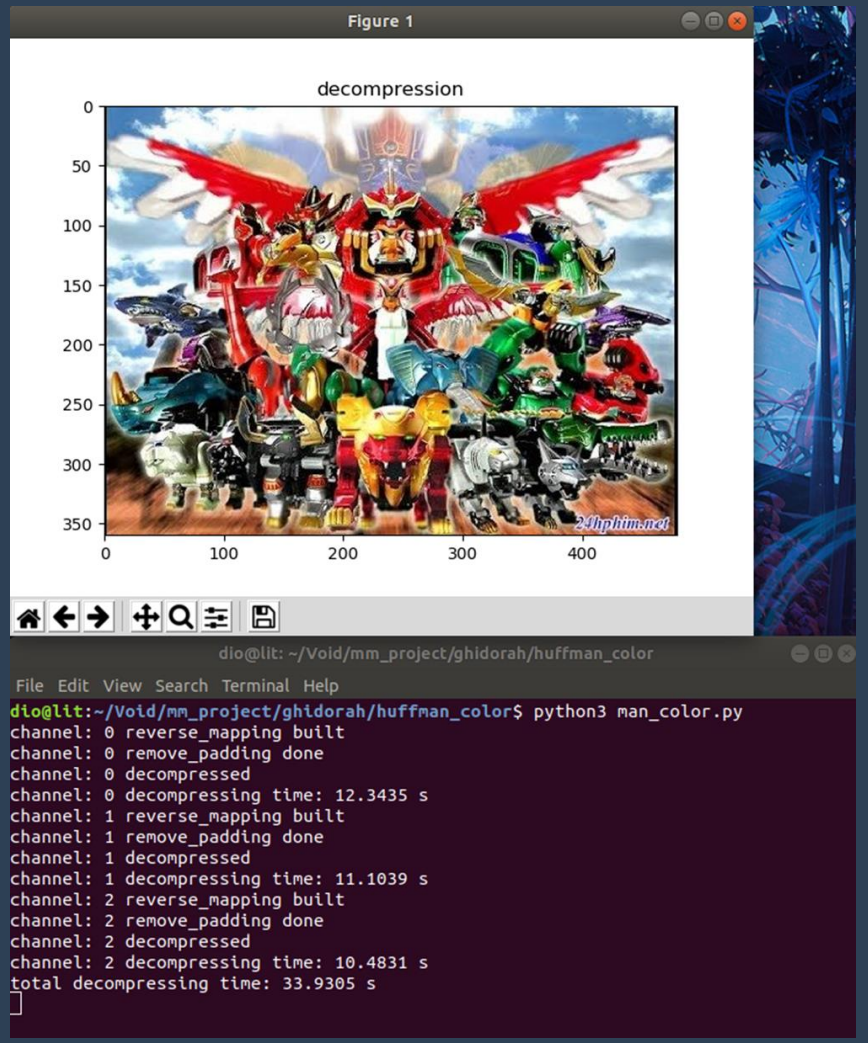
```
70      img = np.reshape(output, (shape[0], shape[
71      return img
72
73
74  if __name__ == "__main__":
75      fCode, fKey, fValue, fShape = ReadFile()
76      code, dictionary, value = list(), list(),
77      shape = ReadFileToShapeImage(fShape)
78      total = 0
79      for i in range(3):
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\goku\Documents\Python Scripts\TTDPT> cd 'c:\Us
1'; & 'C:\Users\goku\Anaconda3\python.exe' 'c:\Users\goku\
t' '--host' 'localhost' '--port' '51634' 'c:\Users\goku\Dc
Done decode channel 0 : --- 3.48 seconds ---
Done decode channel 1 : --- 3.11 seconds ---
Done decode channel 2 : --- 3.17 seconds ---
Done decode: --- 9.76 seconds ---
```

hello





Original size: 518400 bytes
total compression size: 512728 bytes
total compressing time: 0.5022s

Total decompressing time: 33.9305s

```
Image: sieunhan.jpg
channel: 0 keys written
channel: 0 values written
channel: 0 encoded
Done encode channel 0 : --- 4.34 seconds ---
channel: 1 keys written
channel: 1 values written
channel: 1 encoded
Done encode channel 1 : --- 2.16 seconds ---
channel: 2 keys written
channel: 2 values written
channel: 2 encoded
Done encode channel 2 : --- 1.7 seconds ---
shape written
Done encode: --- 8.2 seconds ---
```

```
    for i in range(3):
        start = time.time()
        code.append(ReadFileCodeToList(fCode[i
        dictionary.append(ReadFileDictionaryTo
        value.append(DecodeLZW(code[i], dictio
        end = (time.time()-start).__round__(2)
        total = total + end
        print("Done decode channel "+str(i), "

    output = cv2.merge((value))
    output = cv2.cvtColor(output, cv2.COLOR_BGR
```

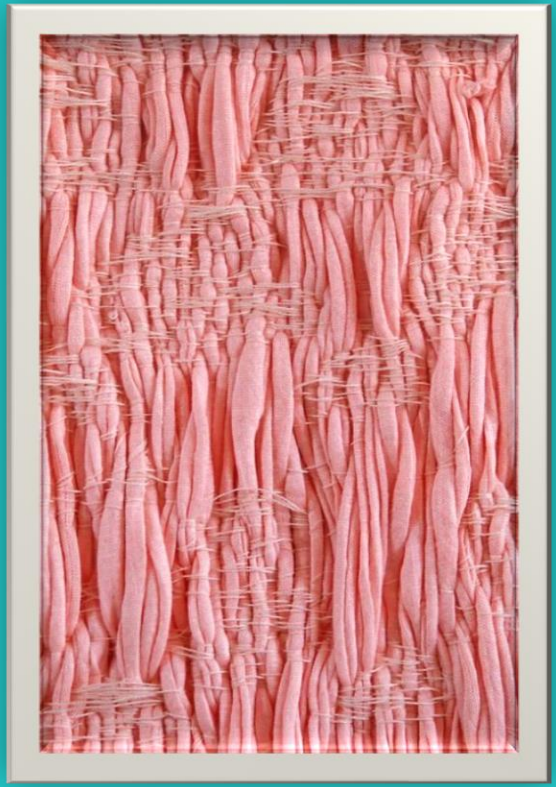
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\goku\Documents\Python Scripts\TTDPT> cd 'c:\Us
1'; & 'C:\Users\goku\Anaconda3\python.exe' 'c:\Users\goku\
t' '--host' 'localhost' '--port' '51260' 'c:\Users\goku\Do
Done decode channel 0 : --- 1.64 seconds ---
Done decode channel 1 : --- 1.74 seconds ---
Done decode channel 2 : --- 1.6 seconds ---
Done decode: --- 4.98 seconds ---
```

COMPARISON AND CONCLUSION

Test 1: color.jpg (980x325) - 955500 bytes

Test 2: lena.jpg
(512x512) - 786432 bytes



Test 3: stripe.jpg
(485x710) - 1033050 bytes

Test 4: tower.jpg
(572x388) - 665808 bytes

COMPRESSION

Test	Original Size (KB)	LZW		Huffman		JPEG	
		Time (s)	Size (KB)	Time (s)	Size (KB)	Time (s)	Size (KB)
1	933	3.4	39	1.82	311	14.6	246
2	768	3.3	1022	2.55	695	3.53	512
3	1009	4.9	1510	3.72	915	6.38	831
4	650	2.9	815	2.13	614	3	424

COMPRESSION

Test	LZW	Huffman	JPEG
1	1 s	180 s	2.5 s
2	2.75 s	55 s	4.8 s
3	4 s	258 s	8.8 s
4	2.1 s	38 s	3.76 s

THANK YOU!