

# Image Compression

Presented by

Sadia Islam Shajne 19CSE032

Rakibul islam 19CSE021

Md Jannatul Ferdhous Emon 19CSE049

Md. Tanbir 19CSE006

Md. Sayfuddin Naeem 17CSE041

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Digital Image Processing

## Image compression

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Image Compression.



Size-270 KB



Size-22 KB

**“Without Compression a CD store only 200 Pictures or 8 Seconds Movie”**

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# What is Image Compression?

Image compression is the process of reducing the file size of an image while preserving as much of its visual quality as possible.

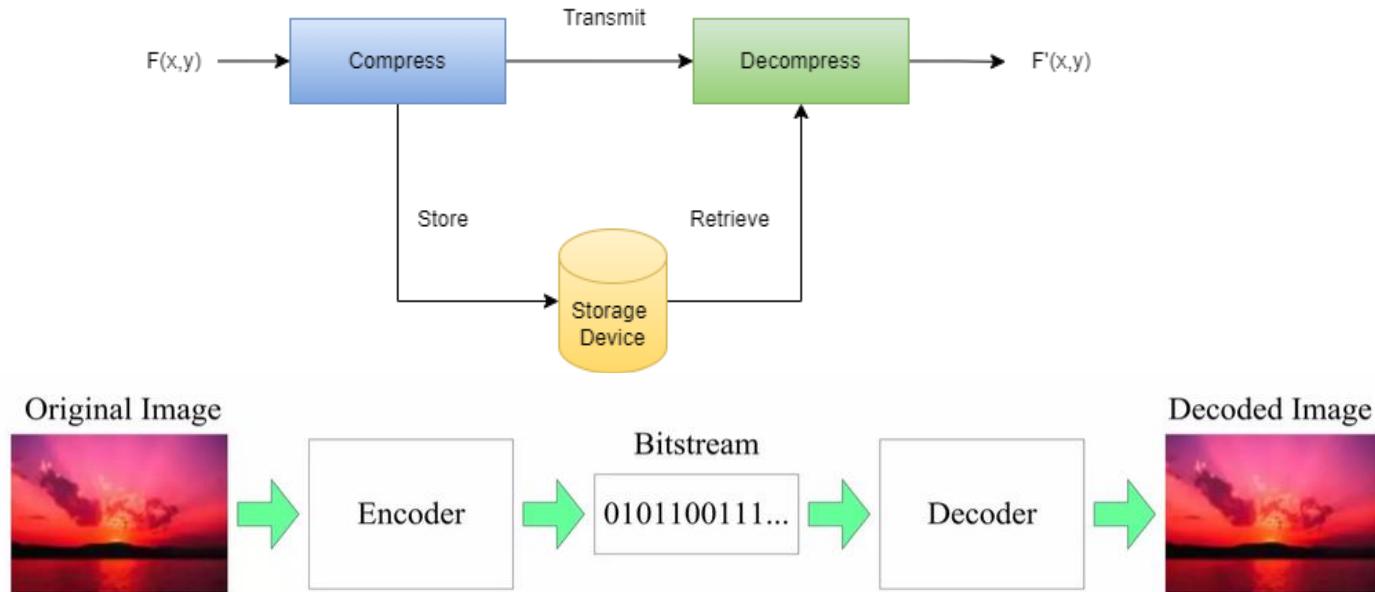
It's crucial for efficient storage, transmission, and display of images, particularly in applications like web browsing, social media, and digital photography.

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Methodology



AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# IMAGE COMPRESSION EXAMPLE

**BEFORE**



**AFTER**



**13.7 MB**

**477 KB**

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Why do we need compression?

- Reduced Storage Requirements
- Faster Transmission

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Applications

- Web and Mobile Applications
- Social Media
- E-commerce
- Digital Photography
- Medical Imaging
- Multimedia

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Types of image compression techniques

1. Lossy Compression

1. Lossless Compression

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Lossy Compression

Lossy compression reduces file size by permanently eliminating certain information, especially redundant or less critical data. This process results in a decrease in image details & quality.

Format: .jpeg photos

Application: widely used for photographs and images on the web.

Advantages: requires less storage, faster in uploading and downloading time.

Disadvantages: loss of details, irreversible loss.

# Lossless Compression

Lossless compression techniques reduce file size without any loss of image quality which means that the original data can be perfectly reconstructed from the compressed data.

Format: .png photos , GIFs.

Application: medical imaging, professional photography, scientific research etc.

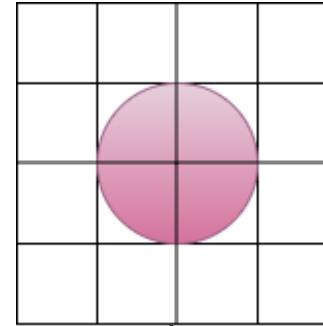
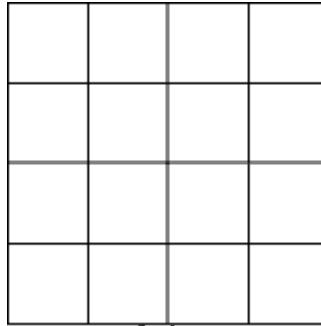
Advantages: can be edited and recompressed multiple times without losing quality.

Disadvantages: large file size, slow transmission rate.

# Data Vs. Information

Data : Consists of the raw pixel values that form an image.

information : Knowledge derived from analyzing and processing the raw data.



Various amounts of data may be used to represent the same amount of information.

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Data Redundancy

- Data contains irrelevant or repeated information called redundant data.
- Image compression techniques leverage this redundancy to minimize the amount of data required to represent an image without substantially affecting its quality.

Let  $n_1$  be the number of data required to represent an information and  $n_2$  be the number of data required to represent that same information after the compression is done.

# Data Redundancy (cont.)

then  $C_R$ , commonly called the Compression Ratio,

$$C_R = \frac{n_1}{n_2}$$

The relative redundancy,  $R_D$  is define as :

$$R_D = 1 - \frac{1}{C_R}$$

# Data Redundancy (cont.)

Let's say a image can be represented with 8 bits, after compression we can represent the same image with 4 bits. Count the compression ratio and the relative data redundancy of it?

$$\text{Compression Ratio, } C_R = \frac{8}{4} = 2$$

$$\text{Data Redundancy, } R_D = 1 - \frac{1}{2} = 0.5 = 50\%$$

## Data Redundancy (cont.)

## Cases that might occur in compression:

If  $n_1 = n_2$ ,  $C_R = 1$  and  $R_D = 0$  no redundancy

If  $n_1 \gg n_2$ ,  $C_B \rightarrow \infty$  and  $R_D \rightarrow 1$       high redundancy

If  $n_1 \ll n_2$ ,  $C_R \rightarrow 0$  and  $R_D \rightarrow \infty$       undesirable

# Huffman Coding - Rakibul Islam

- Huffman coding is a data compression technique that assigns codes to characters in a way that minimises the overall code length. It is named after its inventor, David A. Huffman, who developed the technique in 1952 while working on his Ph.D. thesis at MIT.
- Huffman coding is typically useful for the case where data that we want to compress has frequently occurring characters in it.

# Purpose of Huffman Coding

- Proposed by Dr. David A. Huffman in 1952
  - *“A Method for the Construction of Minimum Redundancy Codes”*
- Applicable to many forms of data transmission
  - Example: text files

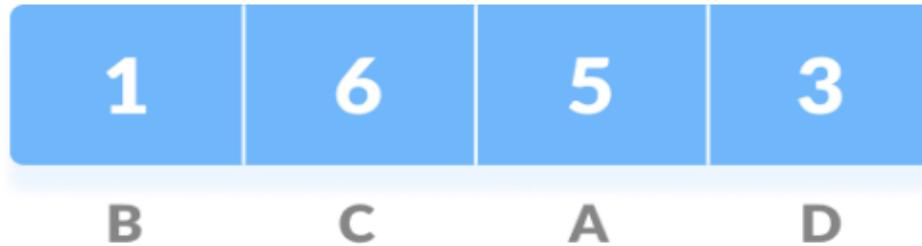
# STEPS TO HUFFMAN CODING

Let assume the string data given below is the data we want to compress -

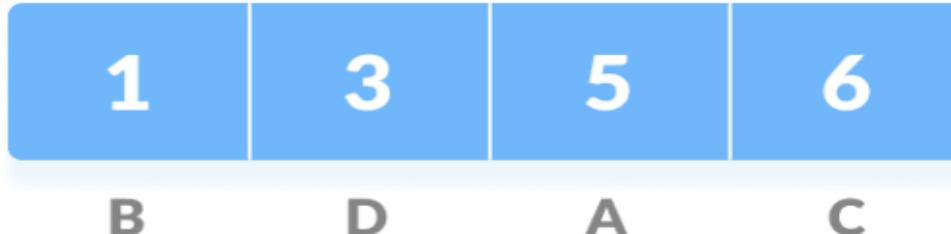


The length of the above string is 15 characters and each character occupies a space of 8 bits. Therefore, a total of 120 bits ( 8 bits x 15 characters ) is required to send this string over a network. We can reduce the size of the string to a smaller extent using Huffman Coding Algorithm. In this algorithm first we create a tree using the frequencies of characters and then assign a code to each character. The same resulting tree is used for decoding once encoding is done.

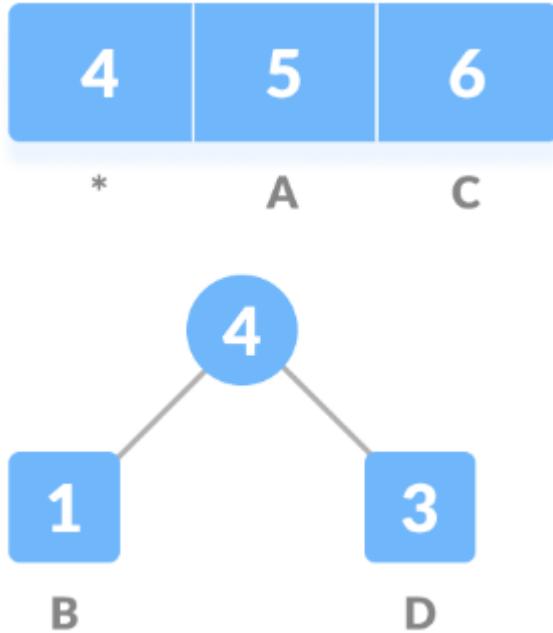
1. First, we calculate the count of occurrences of each character in the string.



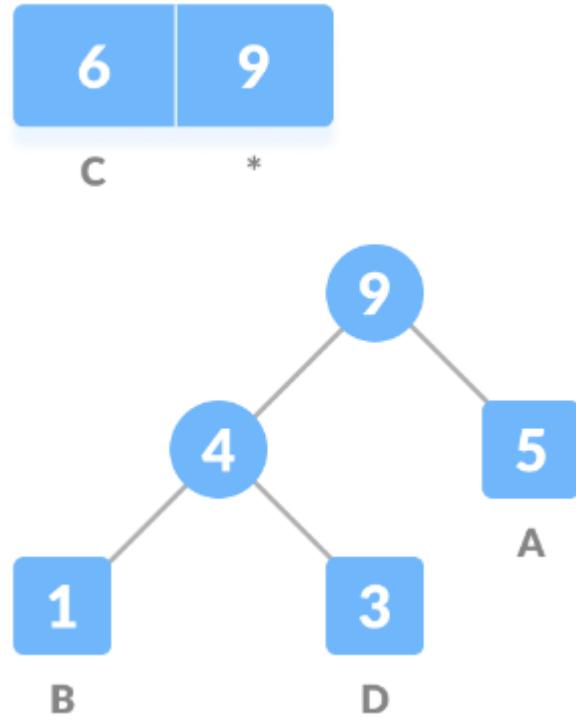
2. Then we sort the characters in the above string in increasing order of the count of occurrence. Here we use PriorityQueue to store.



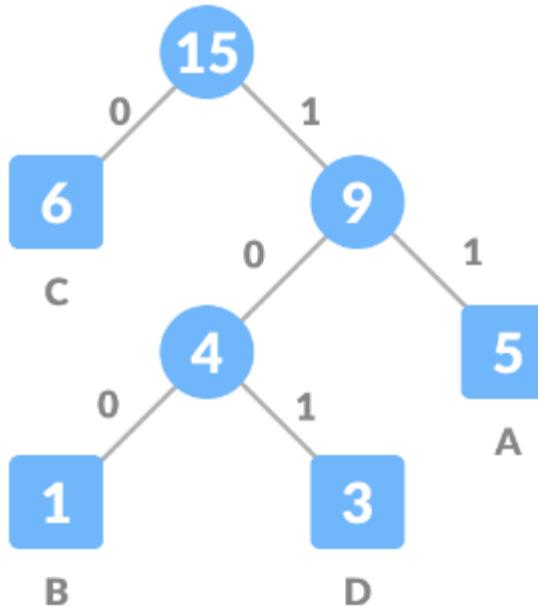
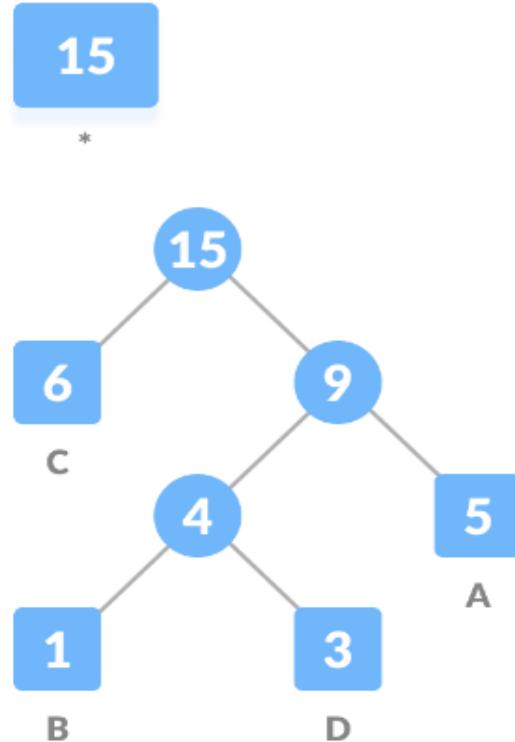
3. Now we mark every unique character as a Leaf Node.
4. Let's create an empty node  $n$ . Add characters having the lowest count of occurrence as the left child of  $n$  and second minimum count of occurrence as the right child of  $n$ , then assign the sum of the above two minimum frequencies to  $n$ .



3. Now remove these two minimum frequencies from Queue and append the sum into the list of frequencies.
4. Add node n into the tree.
5. Just like we did for B and D, we repeat the same steps from 3 to 5 for the rest of the characters ( A and C ). For A -



Repeat for C -



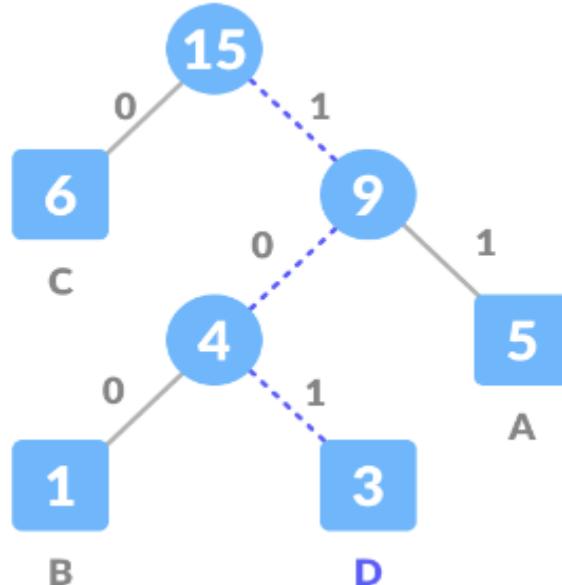
8. We got our resulting tree, now we assign 0 to the left edge and 1 to the right edge of every non-leaf node.

9. Now for generating codes of each character we traverse towards each leaf node representing some character from the root node and form code of it.

Character	Frequency in string	Assigned Code	Size
B	1	<b>100</b>	$1 \times 3 = 3$ bits
D	3	<b>101</b>	$3 \times 3 = 9$ bits
A	5	<b>11</b>	$5 \times 2 = 10$ bits
C	6	<b>0</b>	$6 \times 1 = 6$ bits
$4 \times 8 = 32$ bits	Total = 15 bits		Total = 28 bits

Before compressing the total size of the string was 120 bits. After compression that size was reduced to 60 bits (28 bits + 32 bits).

# STEPS TO HUFFMAN DECODING



To decode any code, we take the code and traverse it in the tree from the root node to the leaf node, each code will make us reach a unique character.

Let assume code 101 needs to be decoded, for this we will traverse from the root as given below -

As per the Huffman encoding algorithm, for every 1 we traverse towards the right child and for 0 we traverse towards the left one, if we follow this and traverse, we will reach leaf node 3 which represents D. Therefore, 101 is decoded to D.

# ALGORITHM

```
1  create and initialize a PriorityQueue Queue consisting of each
2  unique character.
3  sort in ascending order of their frequencies.
4  for all the unique characters:
5      create a new_node
6      get minimum_value from Queue and set it to left child of
7      new_node
8      get minimum_value from Queue and set it to right child of
9      new_node
10 calculate the sum of these two minimum values as
11 sum_of_two_minimum
12 assign sum_of_two_minimum to the value of new_node
13 insert new_node into the tree
14 return root_node
```

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

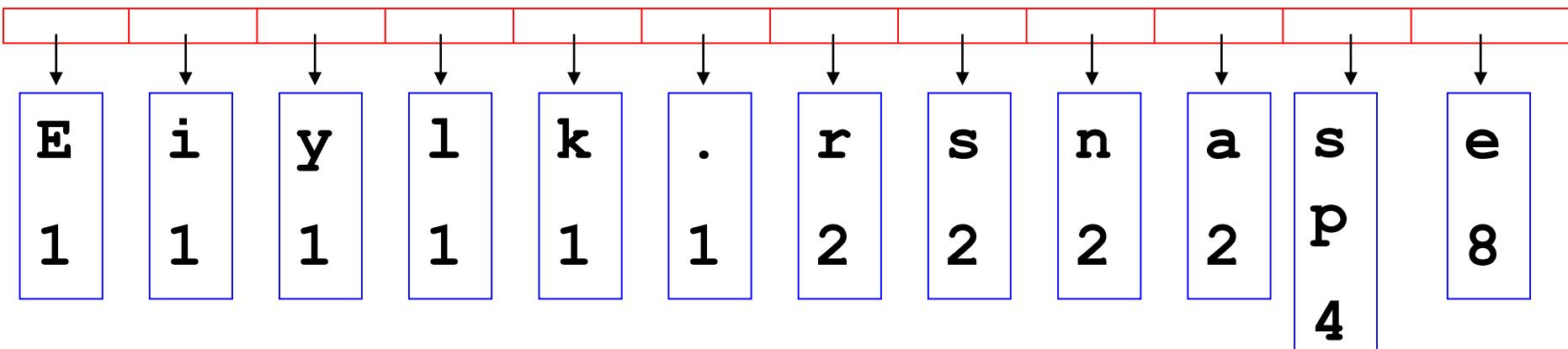
The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

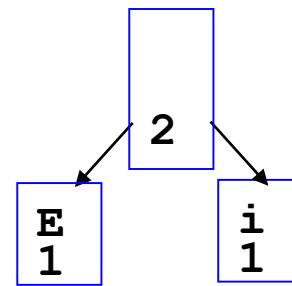
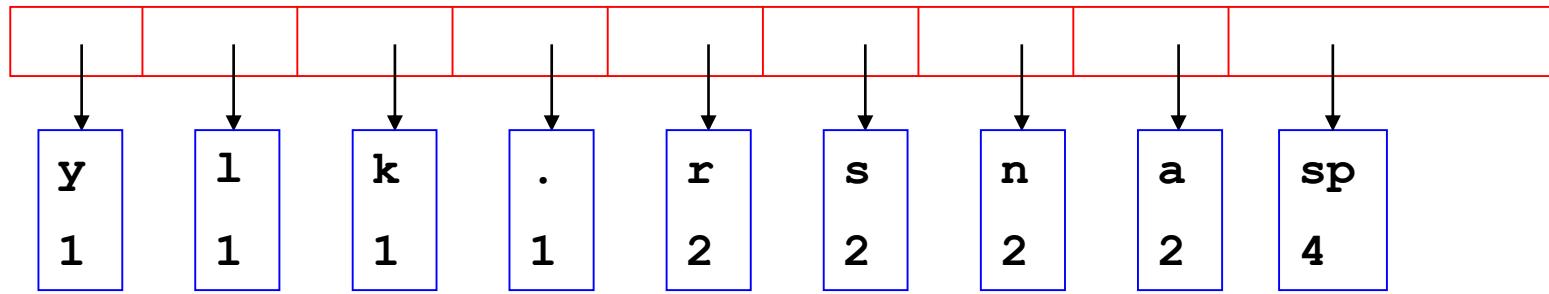
- Huffman coding is a form of statistical coding
- Not all characters occur with the same frequency!
- Yet all characters are allocated the same amount of space
  - 1 char = 1 byte = 8 bit

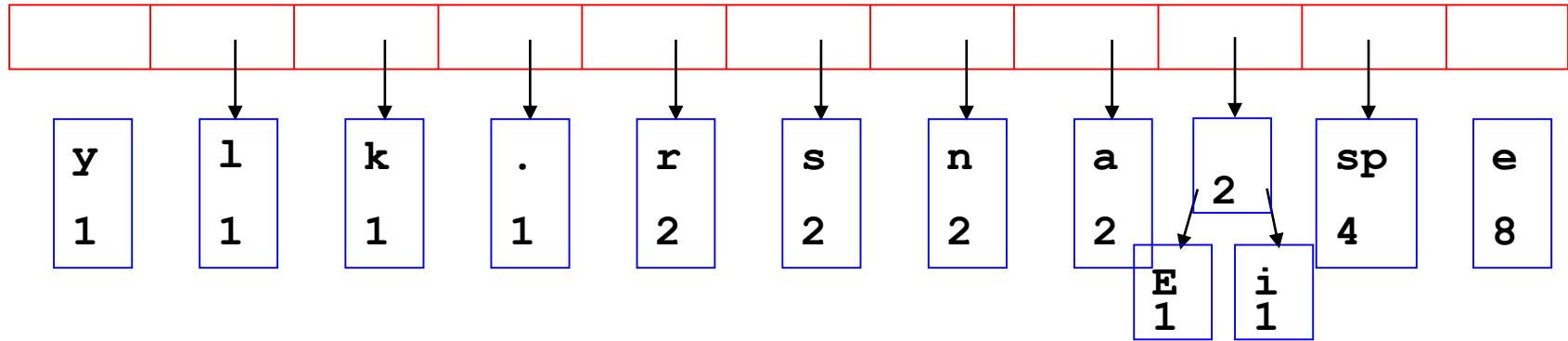
Consider the following short text:

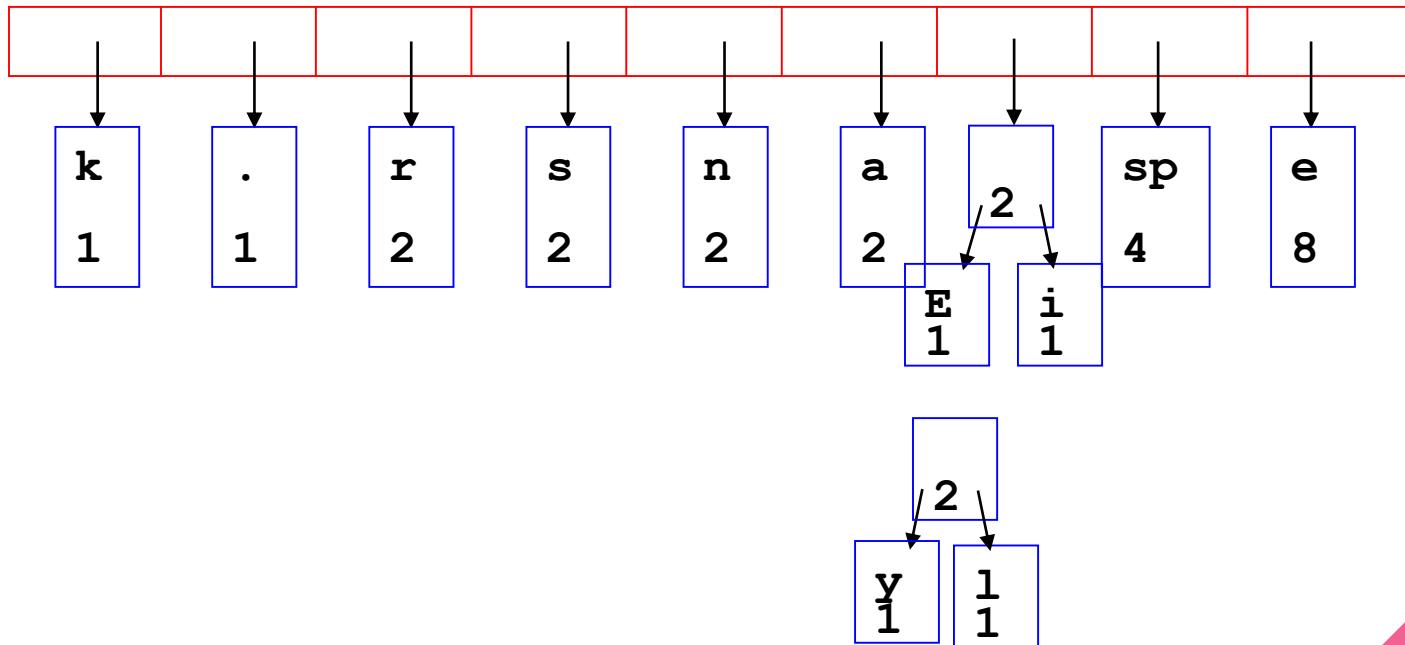
- *Eerie eyes seen near lake.*

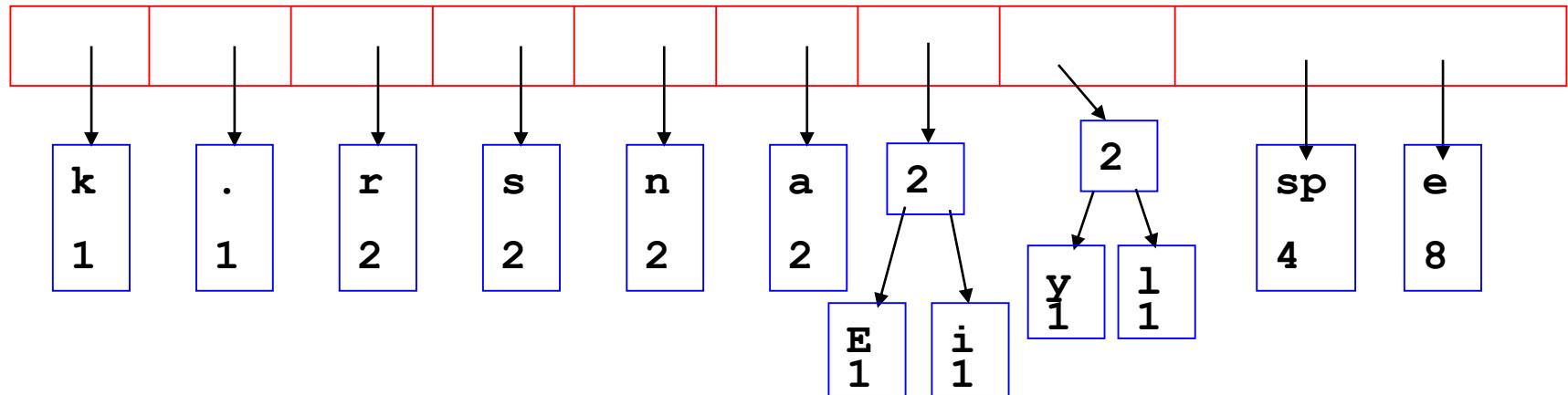
Count up the occurrences of all characters in the text

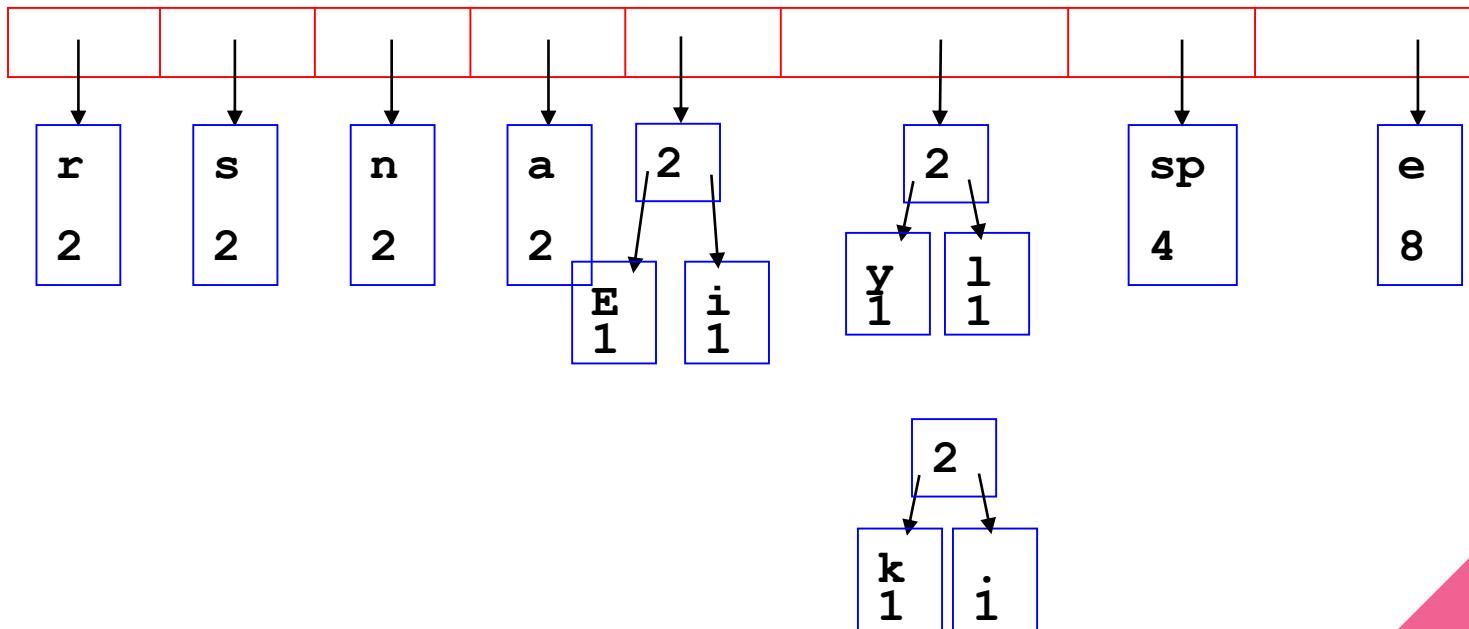


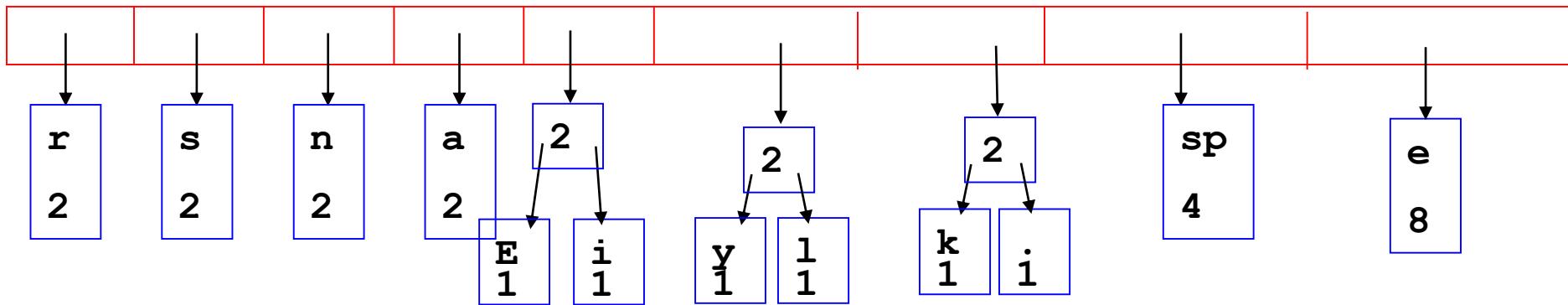


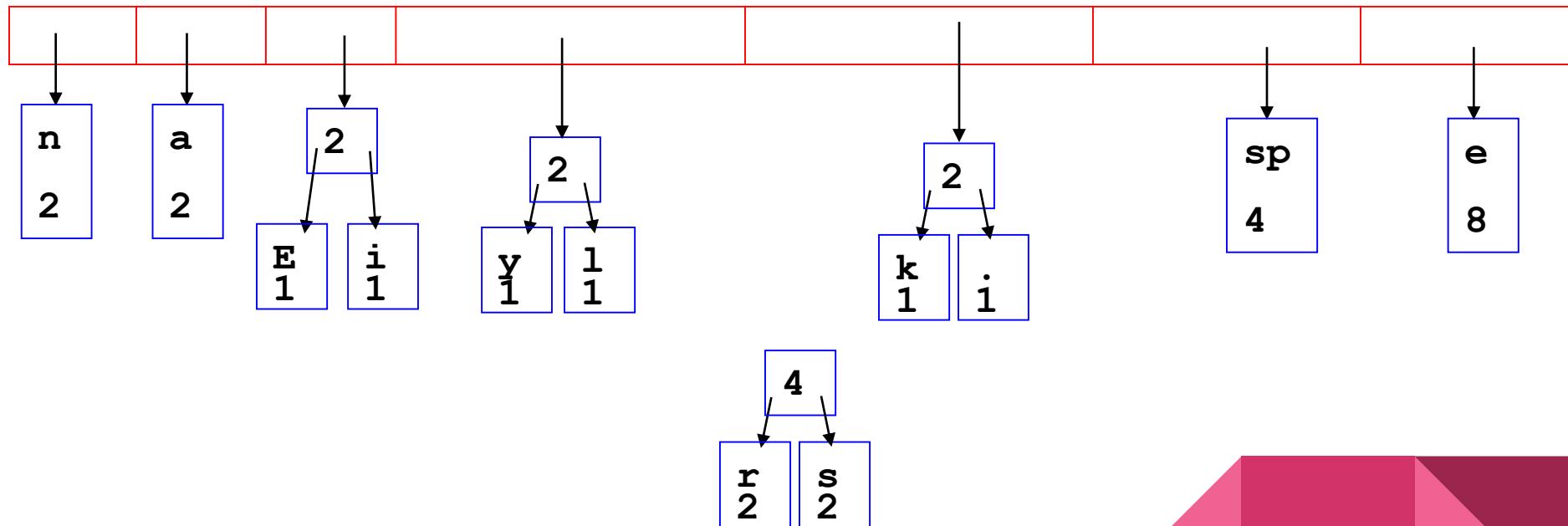


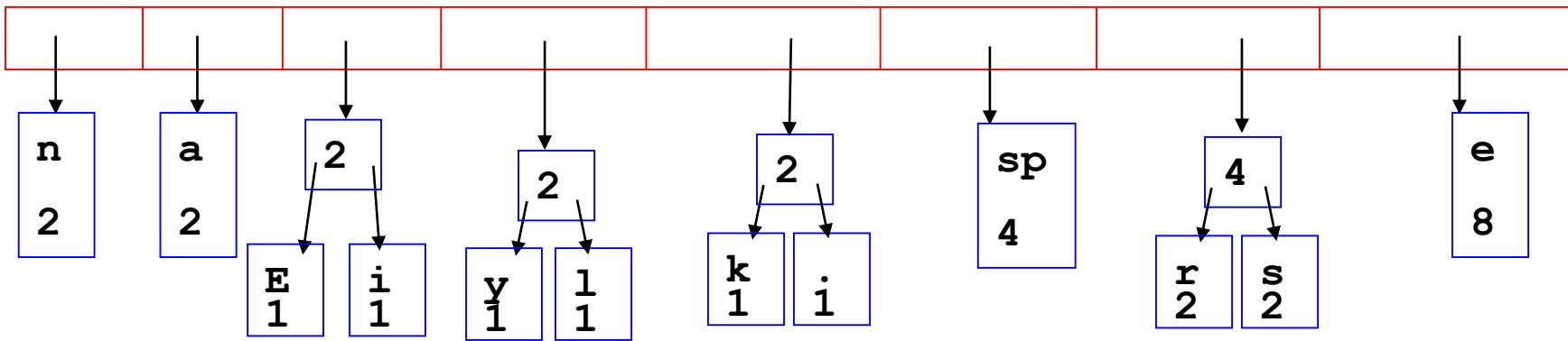


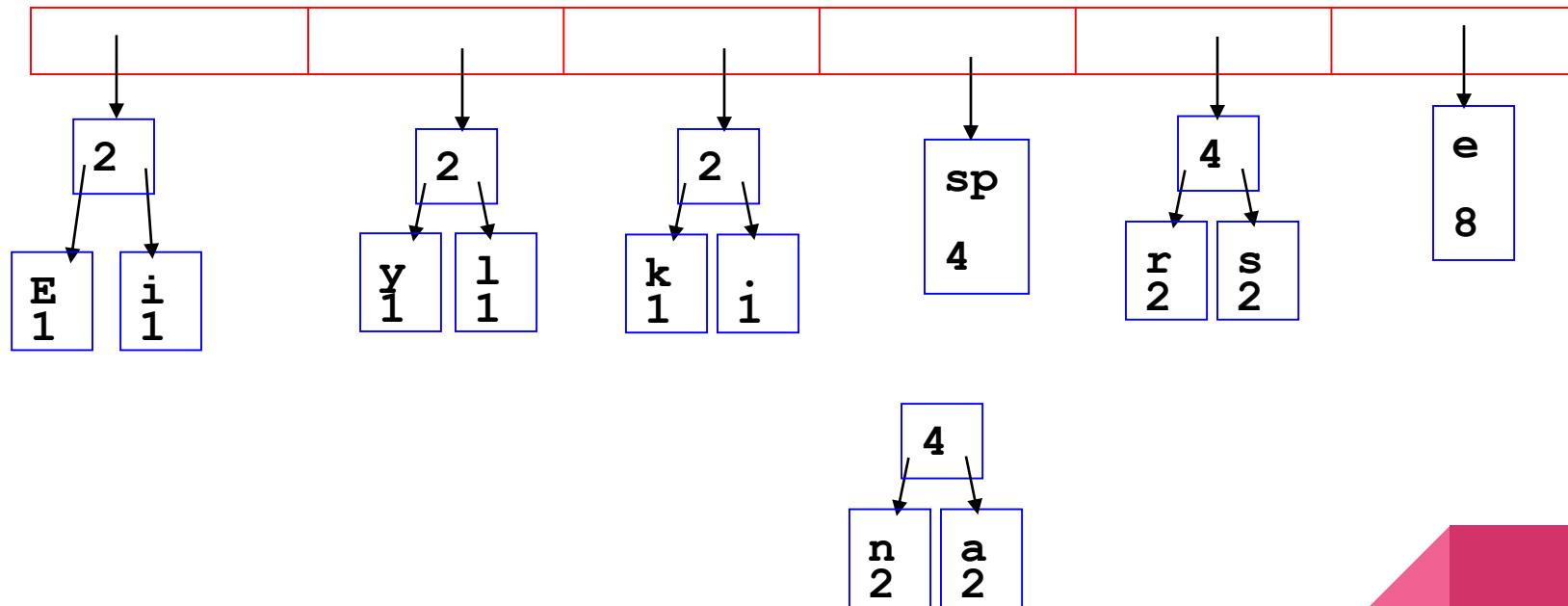


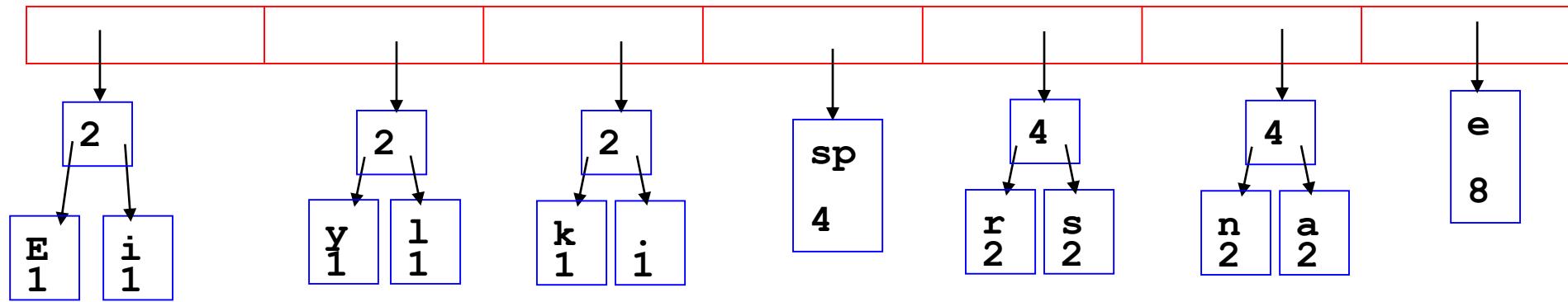


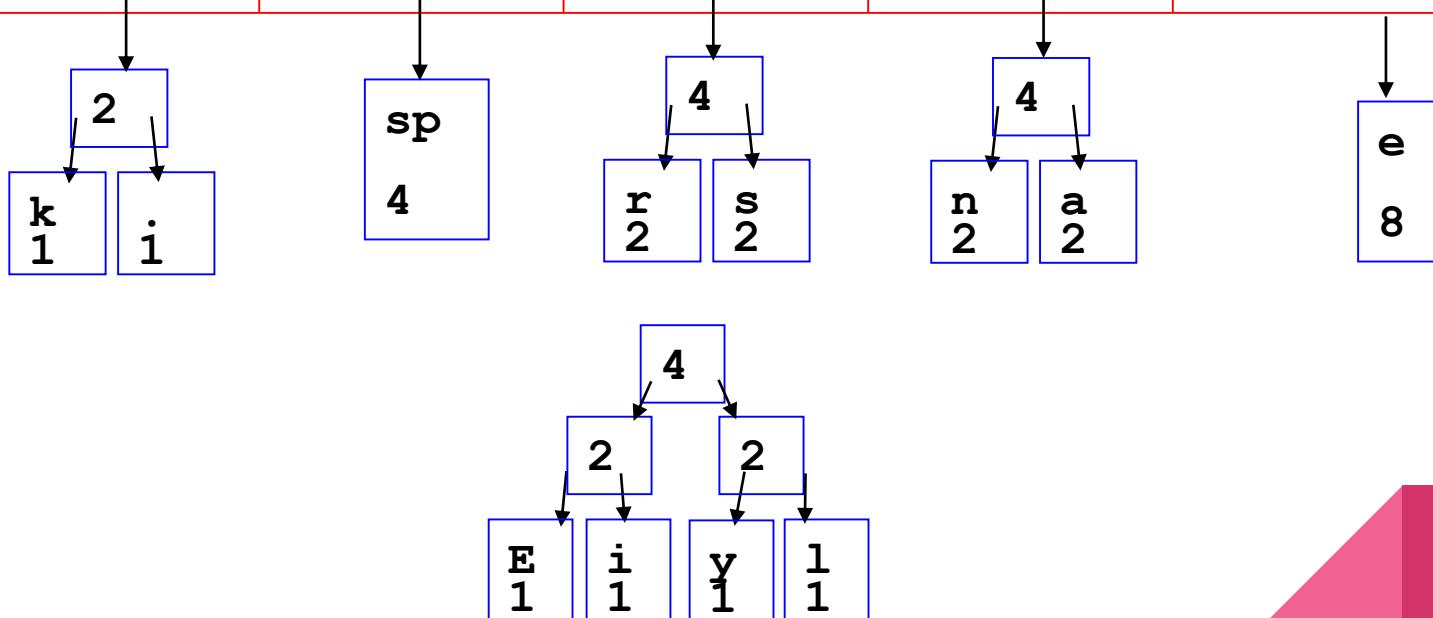


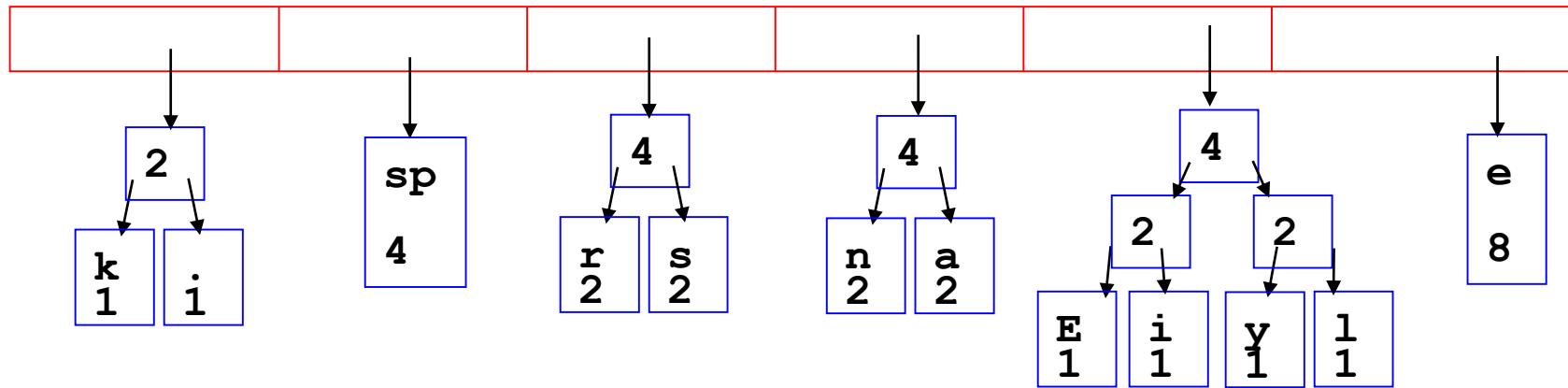


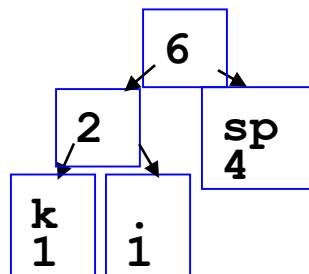
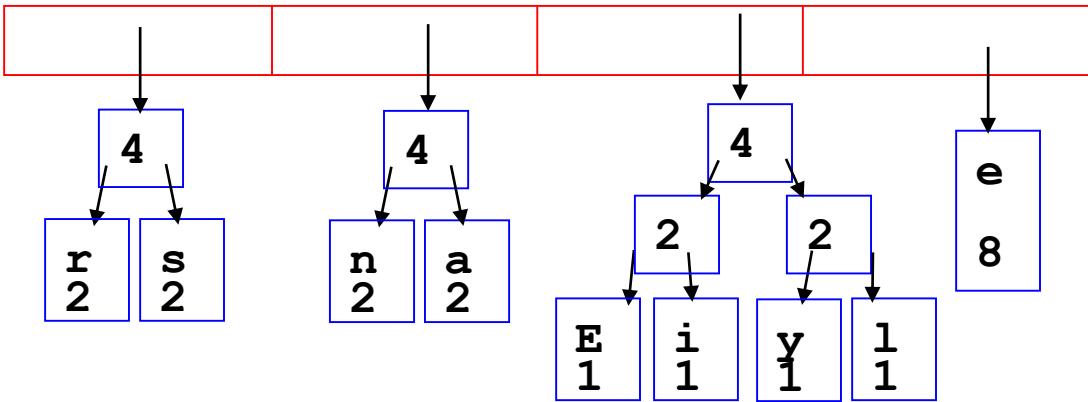








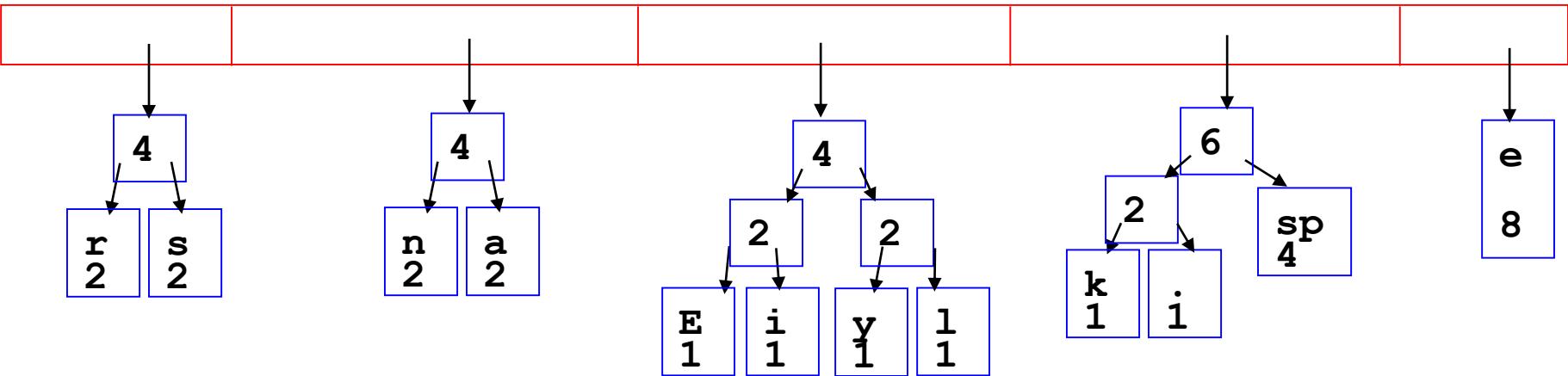


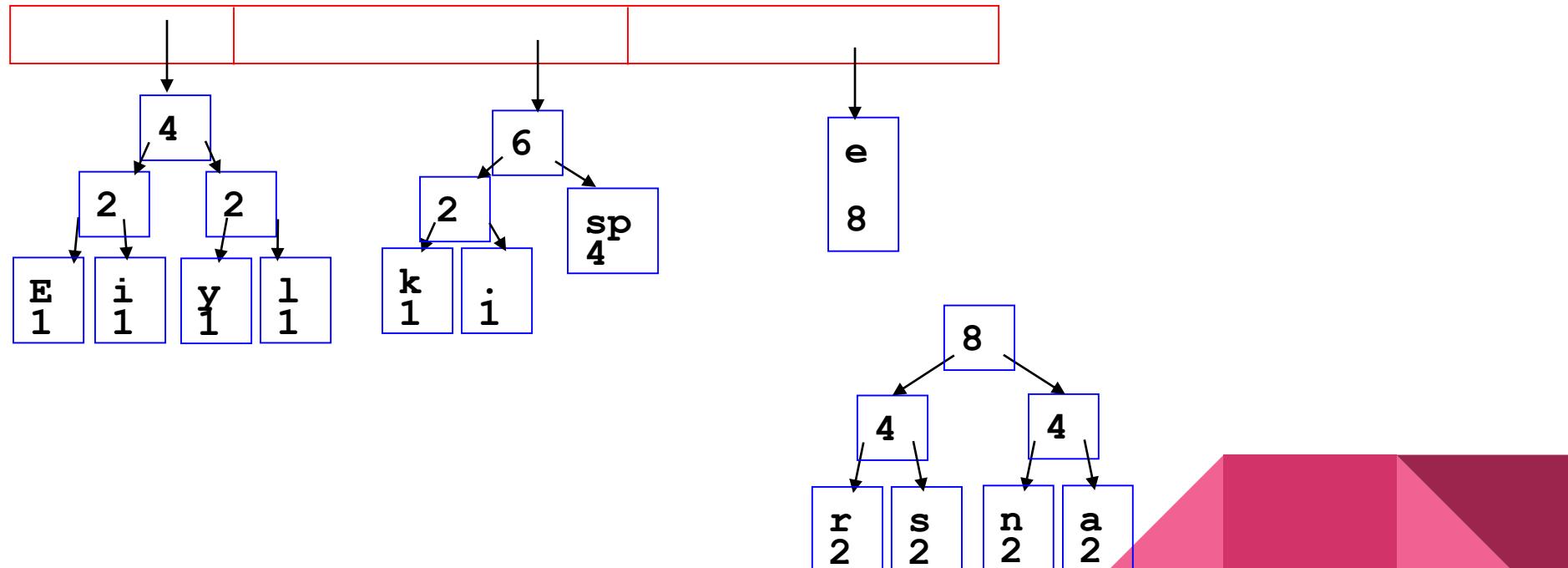


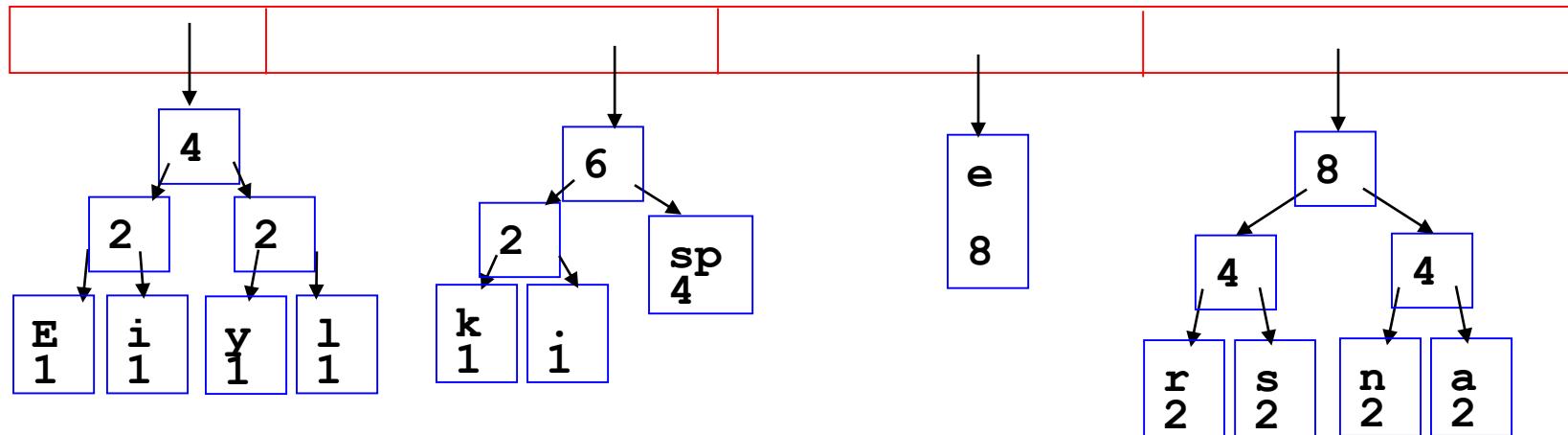
AVAILABLE AT:

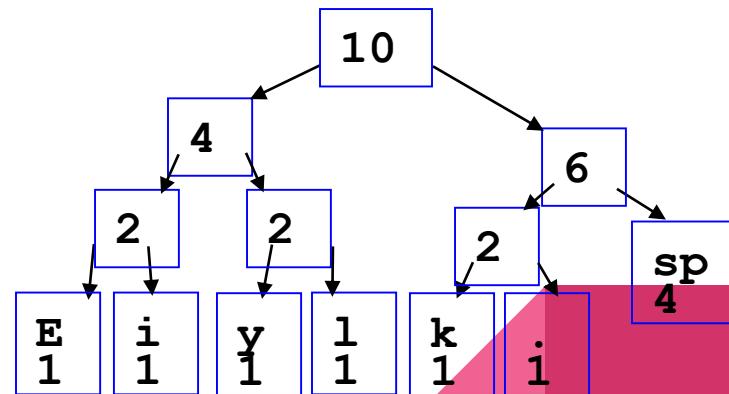
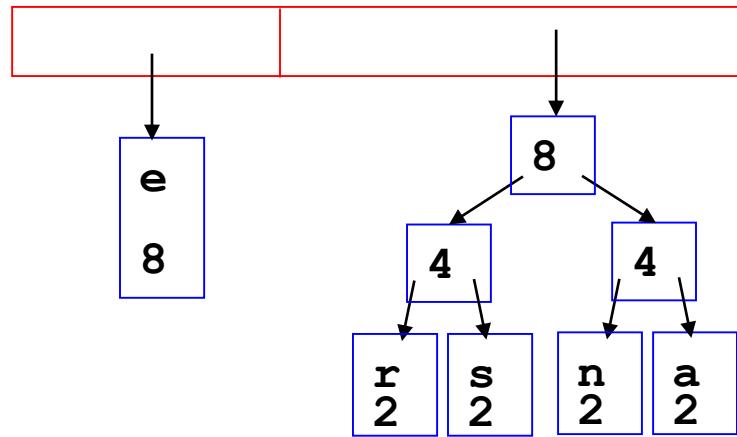
Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))





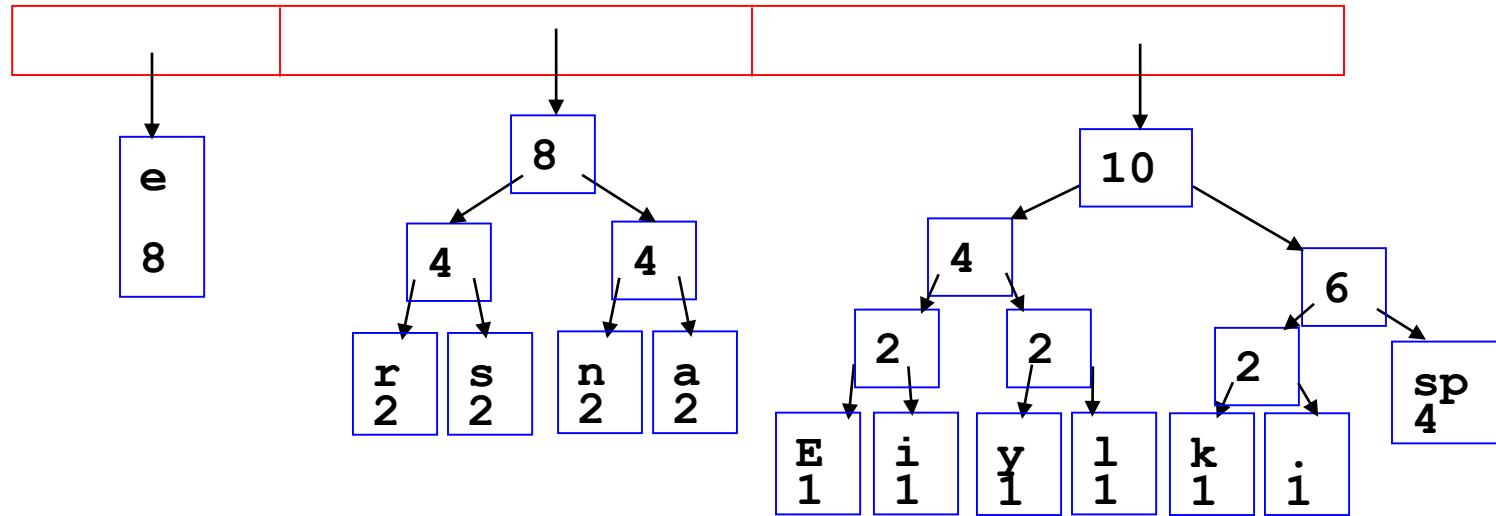


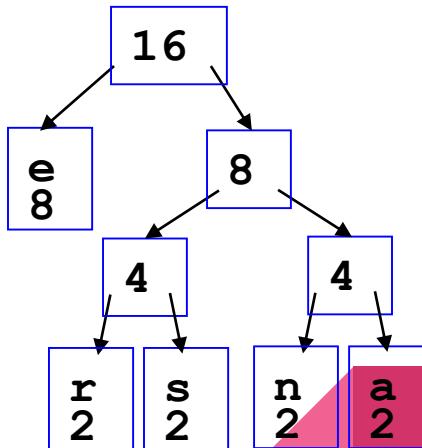
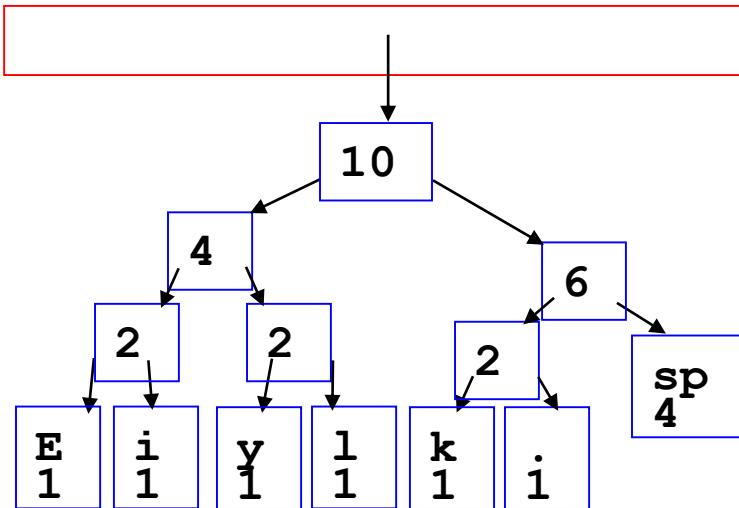


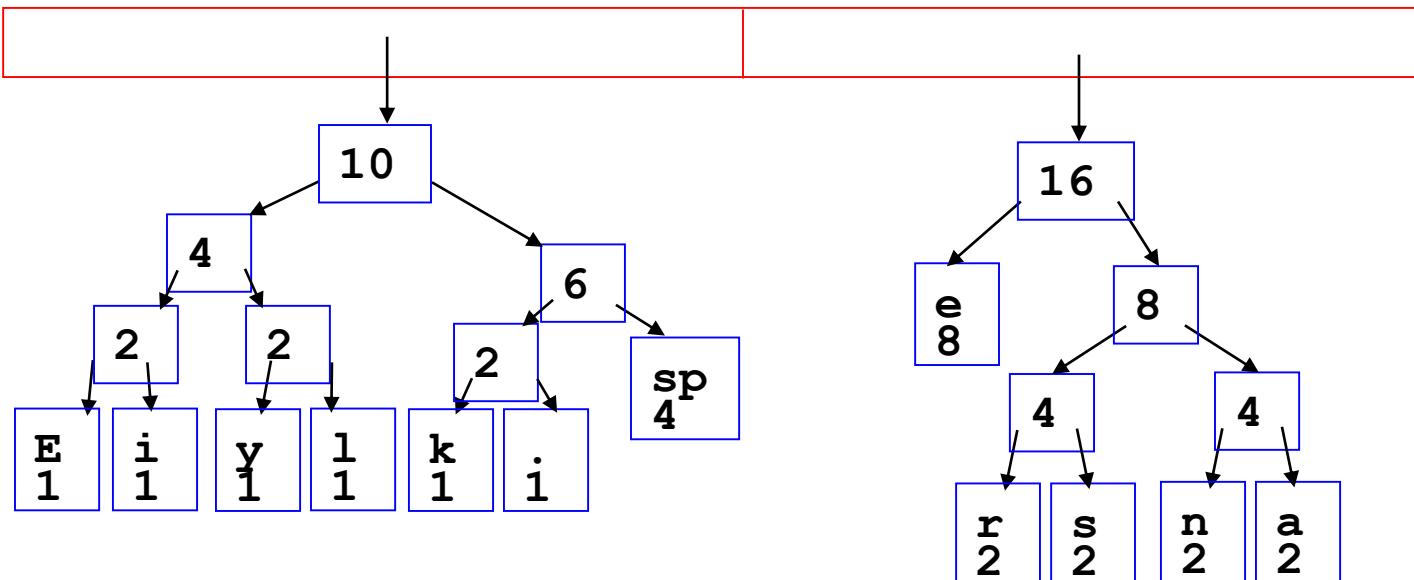
AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))



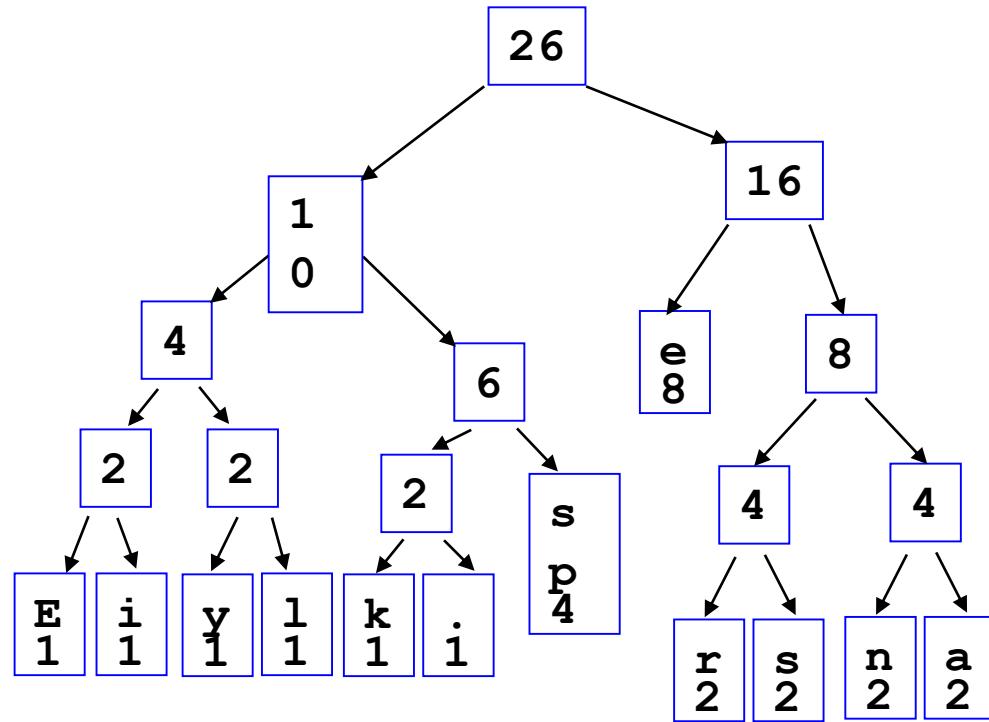




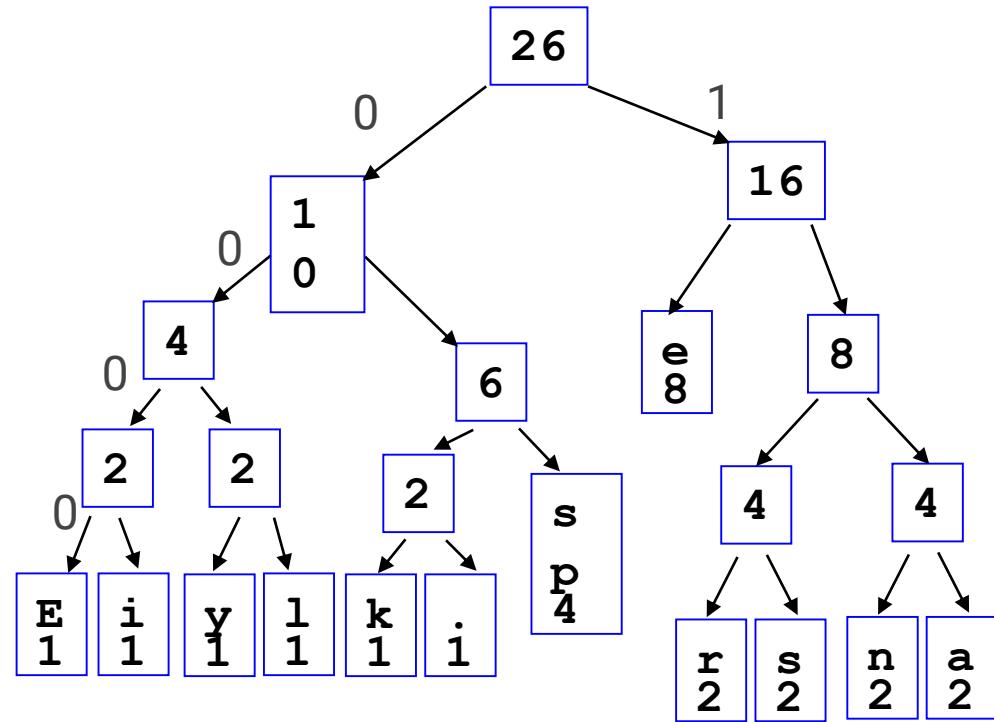
Dequeue the single node left in the queue.

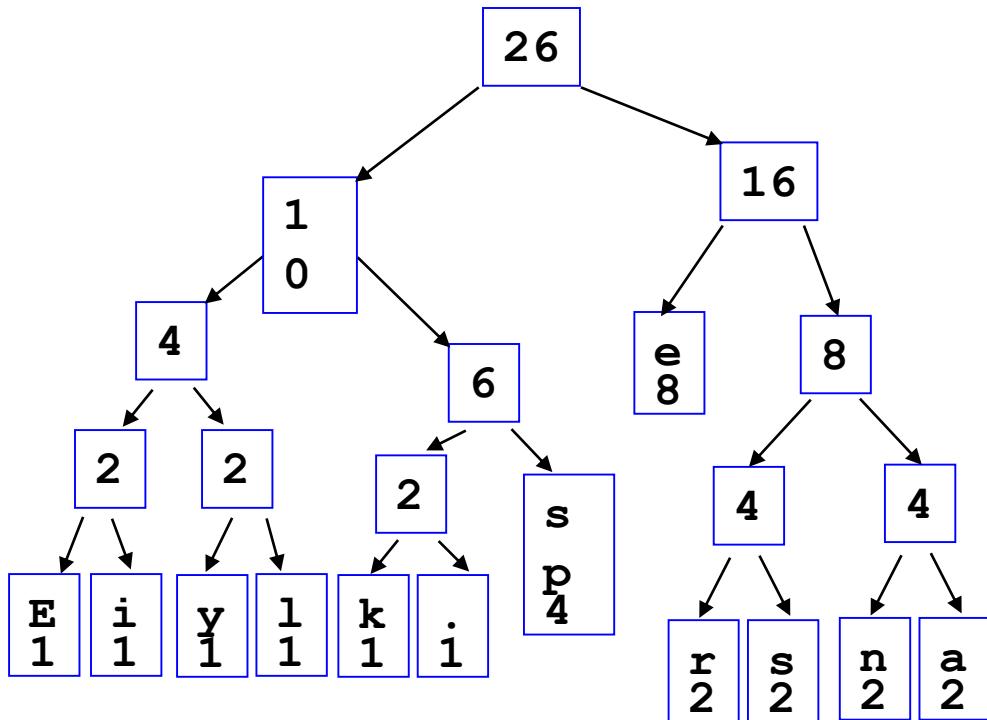
This tree contains the new code words for each character.

Frequency of root node should equal number of characters in text.



- Perform a traversal of the tree to obtain new code words
- Going left is a 0 going right is a 1
- code word is only completed when a leaf node is reached





Char  
E  
i  
y  
l  
k  
.space  
e  
r  
s  
n  
a

Code  
0000  
0001  
0010  
0011  
0100  
0101  
011  
10  
1100  
1101  
1110  
1111

Character	Frequency	Assigned bit	Size
E	1	0000	4
i	1	0001	4
y	1	0010	4
l	1	0011	4
k	1	0100	4
.	1	0101	4
space	4	011	12
e	8	10	16
r	2	1100	8
s	2	1101	8
n	2	1110	8

AVAILABLE AT:

$$\begin{aligned}\text{So total bits} &= \text{number of character} * 8 + \text{total size} \\ &= 13 * 8 + 84 \\ &= 188 \text{ bits}\end{aligned}$$

$$\begin{aligned}\text{But before compress there needed} &= \text{Total number of character} * 8 \\ &= 26 * 8 \\ &= 208 \\ \text{bits}\end{aligned}$$

So the question is where is the benefit of Huffman coding? What is the best and worst case huffman coding?

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Interpixel Redundancy - Emon

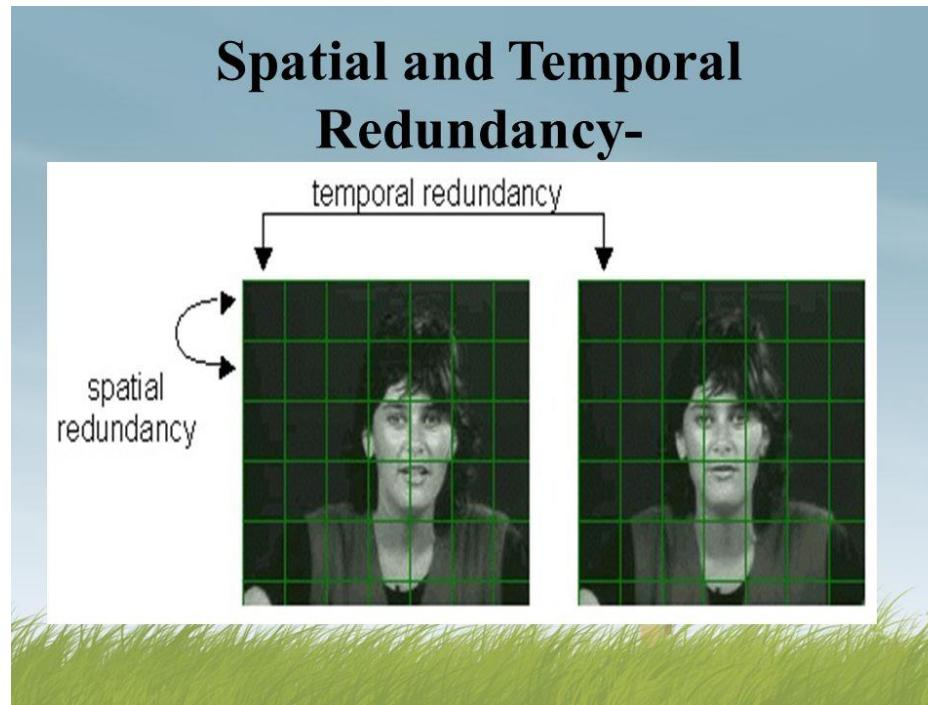
**Interpixel redundancy** is when we can predict a pixel's value by examining its neighboring pixels.

This means that the individual pixels don't hold much special information.

**Types** of Interpixel Redundancy:

- ❖ Spatial Interpixel Redundancy
- ❖ Temporal Interpixel Redundancy

# Interpixel Redundancy



AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Spatial Interpixel Redundancy

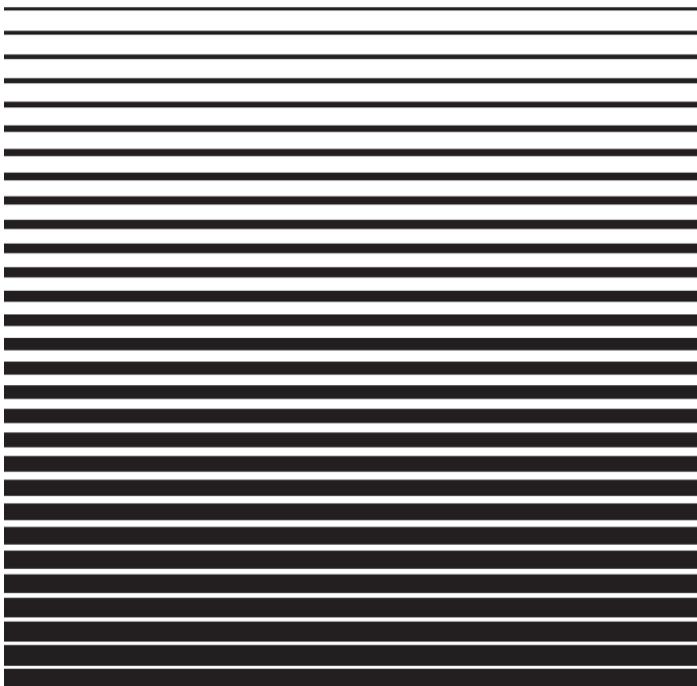
**Spatial Interpixel Redundancy** is due to correlation between the neighboring pixel in an image.

The value of an given pixel can be predicted from the value of its neighboring pixel.

The information carried by individual pixel is relatively small.

To reduce interpixel redundancy the **difference between adjacent** can be used to represent an image.

# Spatial Interpixel Redundancy



AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Temporal Interpixel Redundancy

**Temporal Interpixel Redundancy** is the statistical correlation between pixels from successive frames in a video sequence.

It is also called **Inter-Frame Redundancy**.

Remove a large amount of redundancy.

Leads to efficient **video compression**.

# Temporal Interpixel Redundancy

Frame 1



Frame 2



Frame 3



Frame 4



Frame 5



AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Algorithms

There is often correlation between adjacent pixels i.e. the value of the neighbors of an observed pixel can often be predicted from the value of the observed pixel.

There are two main **Coding Methods**:

- ❖ Run-Length Coding
- ❖ Difference Coding

# Run-Length Coding

Every code word is made up of a pair (**g, I**); where **g is the gray level** and **I is the number of pixels with that gray level**.

Example:

```
56 56 56 82 82 82 83 80  
56 56 56 56 56 80 80 80
```

Creates the **run-length code** (56, 3) (82, 3) (83, 1) (80, 4) (56, 5)

The code is calculated row by row:



Very efficient coding for binary data.

Important to know position and the image dimensions must be stored with the coded image.

Used in most fax machine.

# Run-Length Coding

0	0	0	0	0	0	0	0	0
0	0	1	1	2	3	3	3	3
0	1	1	3	3	3	4	4	4
0	1	3	3	5	5	4	4	4
0	2	3	3	5	5	5	4	4
0	0	2	3	3	4	6	6	6
0	0	0	2	2	3	4	4	4
0	0	0	0	0	0	0	0	0

gray level	0	1	2	3	4	5	6
#pixels	26	5	5	13	8	5	2

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Run-Length Coding

0	0	0	0	0	0	0	0
0	0	1	1	2	3	3	3
0	1	1	3	3	3	4	4
0	1	3	3	5	5	4	4
0	2	3	3	5	5	5	4
0	0	2	3	3	4	6	6
0	0	0	2	2	3	4	4
0	0	0	0	0	0	0	0

row #	run-length code (gl,rl)
0	(0,8)
1	(0,2), (1,2), (2,1), (3,3)
2	(0,1), (1,2), (3,3), (4,2)
3	(0,1), (1,1), (3,2), (5,2), (4,2)
...	...
7	(0,8)

# Run-Length Coding

row #	run-length code (gl,rl)
0	(0,8)
1	(0,2), (1,2), (2,1), (3,3)
2	(0,1), (1,2), (3,3), (4,2)
3	(0,1), (1,1), (3,2), (5,2), (4,2)
...	...
7	(0,8)

row #	binary code
0	000 111
1	000 001 001 001 010 000 011 010
2	000 000 001 001 011 010 100 001
...	...
7	000 111

# Run-Length Coding

## Compression Achieved:

Original image requires **3 bits per pixel**. So, total required bits =  $8 * 8 * 3 = 192$  bits.

Compressed image has total 29 runs and needs  $3 + 3 = 6$  bits per run. So, total required bits =  $29 * 6 = 174$  bits or **2.72 bits per pixel**.

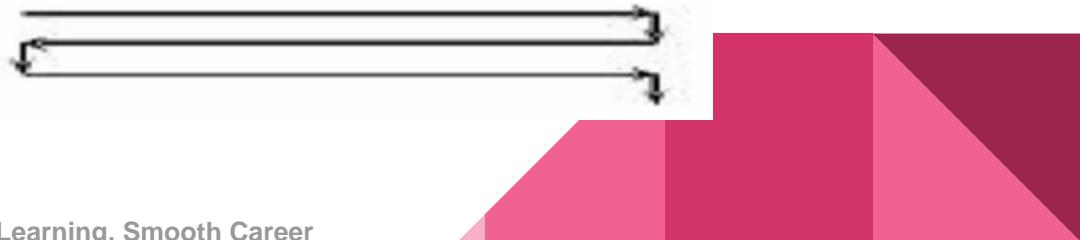
# Difference Coding

$$f(x_i) = \begin{cases} x_i & \text{if } i=0, \\ x_i - x_{i-1} & \text{if } i>0 \end{cases}$$

Example:

- ❖ Original: 56 56 56 82 82 82 83 80 80 80 80
- ❖ Code  $f(x_i)$ : 56 0 0 26 0 0 1 -3 0 0 0

The code is calculate row by row:



# Example of Difference

Original image.

9	8	7	7	7	5	5	5
7	7	7	7	4	4	5	5
6	6	6	9	9	9	6	6
6	6	7	7	7	9	9	9
3	7	7	8	8	8	3	3
3	3	3	3	3	3	3	3
10	10	11	7	7	7	6	6
4	4	5	5	5	2	2	6

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career  
The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

Difference image.

9	-1	-1	0	0	-2	0	0
0	0	0	3	0	-1	0	0
-1	0	0	3	0	0	-3	0
0	-1	0	0	-2	0	0	3
-3	4	0	1	0	0	-5	0
0	0	0	0	0	0	0	0
7	0	1	-4	0	0	-1	0
0	-1	0	0	3	0	-4	0

Tanbir - 19CSE006

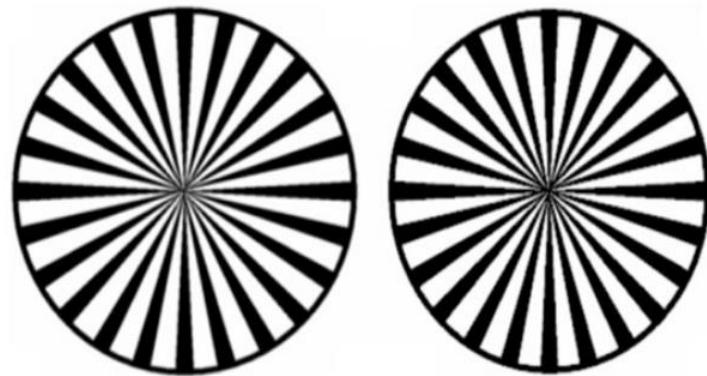
# What is Psychovisual Redundancy ???

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Is there any difference between this two images?



Here, Figure(Left) depicts an image with 256 possible **gray levels** of 8-bit image and Fig.(Right) depicts the same image after uniform **quantization of 1 bit** or 128 possible gray levels.

The two images can look **identical** to a human observer, however, **quantization** is performed on the second figure.

AVAILABLE AT:

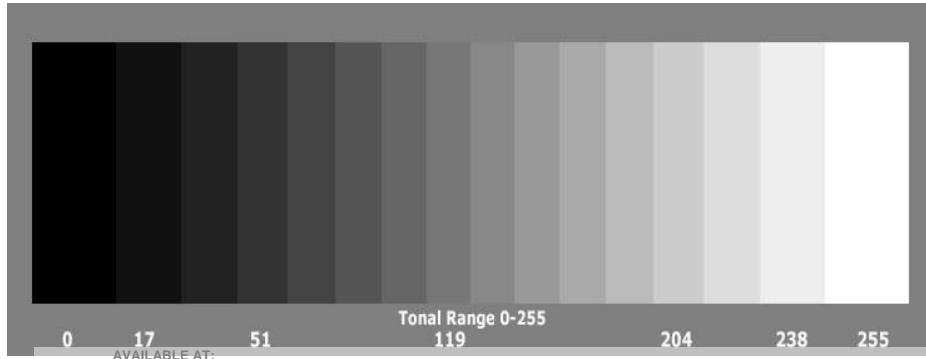
Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

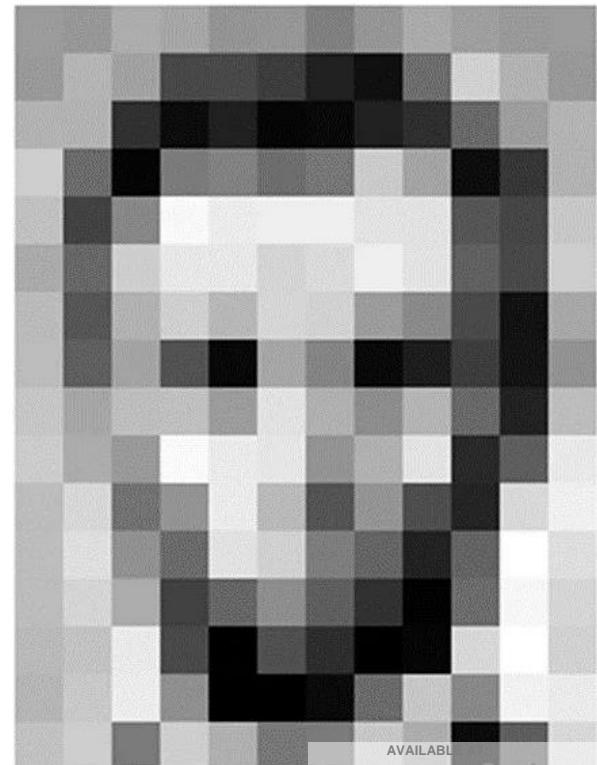
# What is Gray Levels or Gray Scale?

**Gray level** in the context of digital image processing refers to the value that represents the **intensity of light** at a pixel in a grayscale image.

In a grayscale image, each pixel's value ranges from black to white, typically on a scale from 0 to 255, where **0 represents black**, **255 represents white**, and intermediate values represent varying shades of gray.



# What is Gray Levels or Gray Scale?



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	6	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	105	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	85	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	95	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	10	96	218

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	105	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	85	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	95	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

# What is Psychovisual Redundancy? (irrelevant information)

**Psychovisual redundancy** refers to parts of an image that can be altered or removed without a noticeable change in the visual quality as perceived by humans.

The elimination of psychovisual redundancy is termed as **quantization** and it is an **irreversible** process.

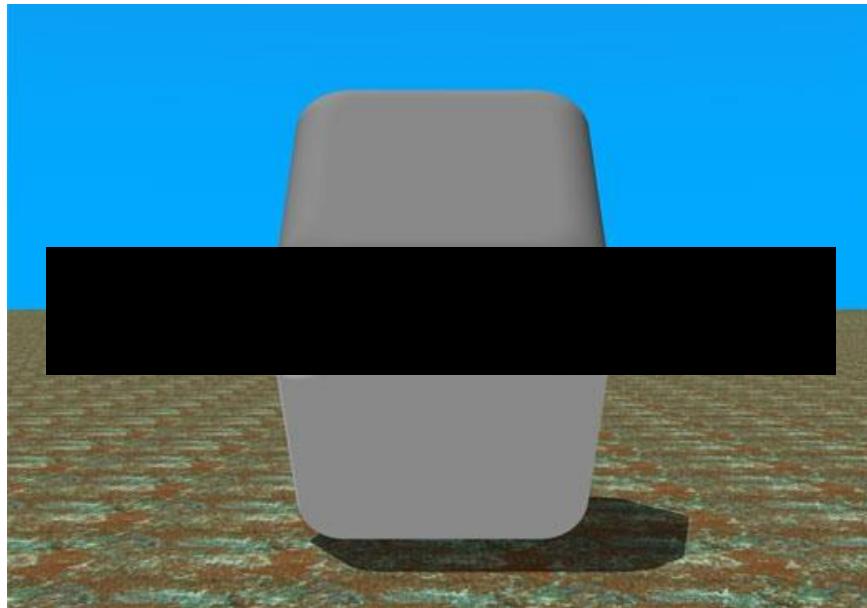
AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Why such name “Psycho-visual”??

What is the color of these two parts?? What you can see?



AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Psychovisual Redundancy (Example-02)

In Fig.1 shows a monochrome image with 256 gray levels or **Eight** bit.

Fig.2 is the same image after **uniform quantization** to **four bits** or 16 possible levels.

The resulting compression ratio is **2 : 1**



**Note that** false contouring is present in the previously smooth regions of the original image. This is the natural visual effect of more coarsely representing the gray levels of the image.

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Uniform Quantization

**Uniform quantization** refers to the process of converting a continuous range of gray levels (or color intensities) in an image into a finite number of discrete levels where the spacing between these levels is uniform.

## Key Concepts of Uniform Quantization in Digital Image Processing:

**Uniform Step Size:** The difference between consecutive quantization levels is constant. This step size determines the resolution of the quantized image.

**Gray Level Quantization:** Each pixel's intensity in a grayscale image is mapped to the nearest quantization level based on the uniform step size.

**Quantization Levels:** The number of distinct intensity values to which the continuous range is mapped. For example, reducing an 8-bit image (256 levels) to 4 levels

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Uniform Quantization

Suppose we have an 8-bit (256 levels) grayscale image (with pixel values ranging from 0 to 255) and we want to quantize it into 4 levels.

## 1. Determine the Step Size:

- The range of pixel values is  $255 - 0 = 255$ .
- For 4 levels, the step size  $\Delta$  is  $\frac{256}{4} = 64$ .

## 2. Quantization Levels:

- The levels could be represented as the midpoints of the quantization intervals: 32, 96, 160, 224.

## 3. Quantization Intervals:

- $[0, 64) \rightarrow 32$
- $[64, 128) \rightarrow 96$
- $[128, 192) \rightarrow 160$
- $[192, 256) \rightarrow 224$

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Psychovisual Redundancy (Example-02)

Fig.3 shows the improvements **from quantization** leveraging the human visual system.

Although the compression resulting from this quantization also is **2 : 1**, false contouring is greatly reduced here.

The method used is called **improved gray-scale (IGS)** quantization.



**Note that** in either case (Fig.2 & Fig.3), decompression is both unnecessary and impossible (as quantization is an irreversible operation).

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Improved Gray Scale (IGS) quantization

**IGS (Improved Gray Scale)** quantization is a technique used in image processing to reduce the number of bits needed to represent pixel values while preserving the visual quality of the image.

- Set the initial previous IGS code to zero. Then for Each Pixel:
- Compute the sum of the current pixel's gray level and the previous IGS code.
- Derive the current pixel's IGS code from the most significant bits (MSBs) of the sum.
- Update the previous IGS code to be the current IGS code.

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

Pixel	Gray Level	Sum	IGS Code
$i - 1$	N/A	0000 0000	N/A
$i$	0110 1100	0110 1100	0110
$i + 1$	1000 1011	1001 0111	1001
$i + 2$	1000 0111	1000 1110	1000
$i + 3$	1111 0100	1111 0100	1111

The table describes the process of converting gray levels to IGS codes.

Sayfuddin Naeem 17CSE041

# JPEG COMPRESSION

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Definition

JPEG compression is a widely used method for reducing the file size of digital images, primarily photographs, through a process that balances loss of image quality with file size reduction. It is named after the Joint Photographic Experts Group, the organization that created the standard. JPEG compression involves several steps, combining lossy and lossless techniques to achieve efficient compression

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Importance of JPEG

- ❖ **Reduced File Size:** Significantly decreases image file size for efficient storage.
- ❖ **Faster upload and download:** Accelerates the transfer of images online.
- ❖ **Quicker Image Processing:** Speeds up image processing in applications and software.
- ❖ **Improved Website Performance:** Enhances website loading times and user experience.
- ❖ **Standard for Photographs:** Ideal for compressing photographic images with minimal quality loss.
- ❖ **Quality vs Size Trade-off:** Offers adjustable compression levels for different needs.

# Applications of JPEG

- ❖ Digital Photography
- ❖ Web Images & Social Media
- ❖ Email Attachments
- ❖ Online Shopping & Digital Advertising
- ❖ Photo Storage & Digital Archives
- ❖ Medical Imaging
- ❖ Surveillance
- ❖ Electronic Publishing
- ❖ Photo Printing

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Features of JPEG

- ❖ Lossy Compression
- ❖ High Compression Ratio
- ❖ Color Space Conversion
- ❖ Block-Based Compression
- ❖ Discrete Cosine Transform (DCT)
- ❖ Quantization
- ❖ Entropy coding
- ❖ Adjustable Quality
- ❖ Wide Compatibility

AVAILABLE AT:

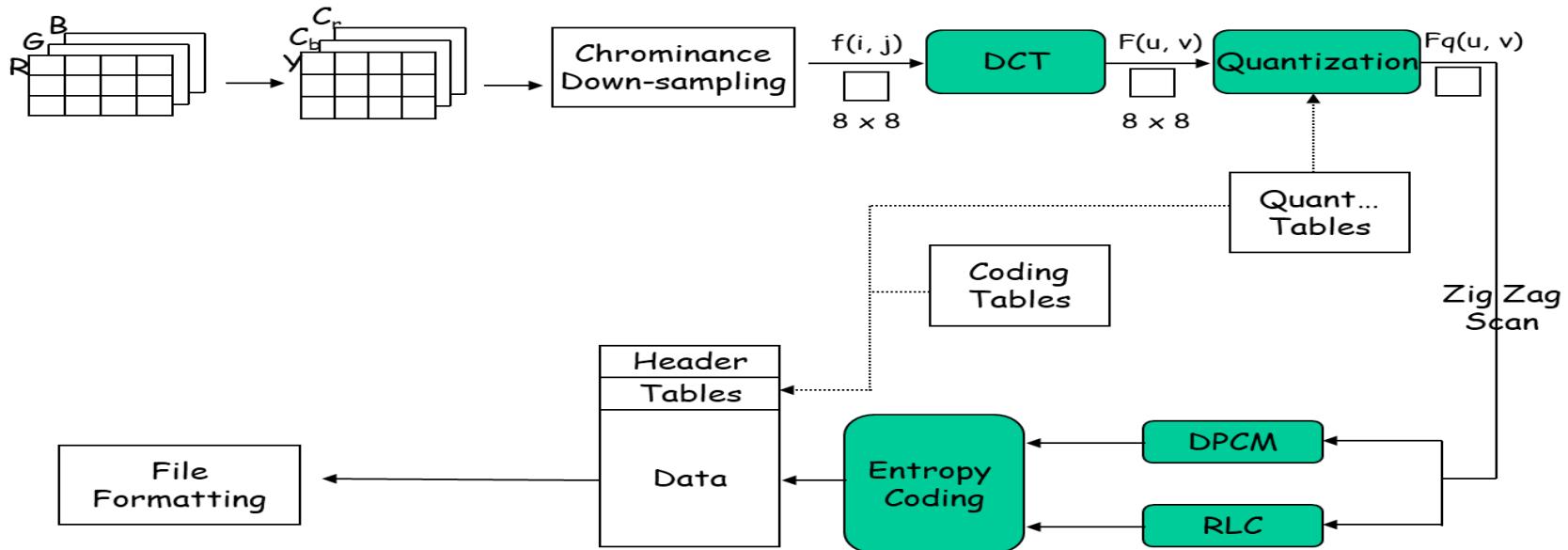
**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Steps of JPEG Compression

1. Splitting input image into 8X8 block
2. Convert the image to RGB to YCbCr color space
3. Reduce the resolution of chrominance components (Cb & Cr)
4. Divide the image into 8X8 pixel blocks
5. Discrete Cosine Transform of each 8x8 pixel array  $f(x,y) \rightarrow F(u,v)$
6. Quantization using a table or using a constant
7. Zig-Zag scan to exploit redundancy
8. Differential Pulse Code Modulation(DPCM) on the DC component and Run length Coding of the AC components
9. Entropy coding (Huffman) of the final output
10. Organize Compressed data into JPEG file format with necessary metadata

# Steps visualization



AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Splitting image

The input image is divided into smaller block having  $8 \times 8$  dimensions, summing up to 64 units in total. Each of this unit is called a *pixel*, which is the smallest unit of any image.



Original Image



Image splitting into  
multiple  $8 \times 8$  pixel blocks

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# YCbCr Color Model

- ❖ Y: The luminance of the image which represents the brightness.
- ❖ Cb: The chrominance of the image which represents the difference between the gray and blue.
- ❖ Cr: The chrominance of the image which represents the difference between the gray and red.

# RGB to YCbCr Conversion



8X8 pixel  
1 pixel = 3 components

MCU with sampling  
factor (1, 1, 1)

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.334 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix}$$

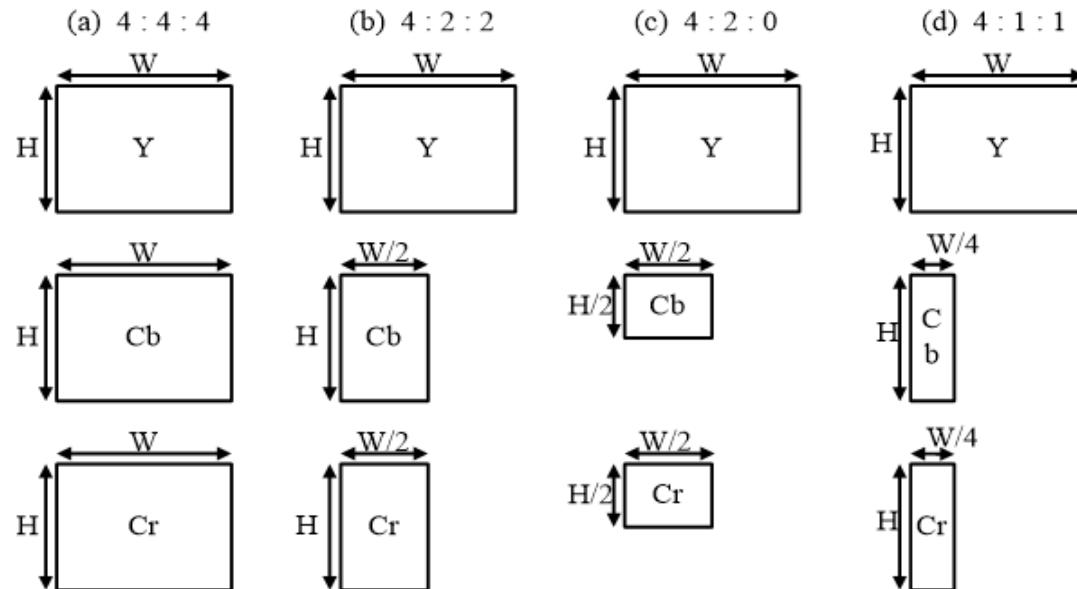
AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Chrominance Reduction

- ❖ The name of the format is not always related to the subsampling ratio



AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Discrete Cosine Transform (DCT)

- ❖ The DCT uses the cosine function, therefore not interacting with complex numbers at all.
- ❖ DCT converts the information contained in a block (8X8) of pixels from ***spatial*** domain to ***frequency*** domain.

Why DCT?

- ❖ Human vision is insensitive to high frequency components, due to which it is possible to treat data corresponding to high frequencies as redundant. To segregate the raw image data on the basis of frequency, it need to be converted into frequency domain, which is the primary function of DCT.

# DCT Formula

## ❖ 1-D DCT –

$$F(w) = \frac{a(u)}{2} \sum_{n=0}^{N-1} f(n) \cos \frac{(2n+1)w\pi}{16}$$

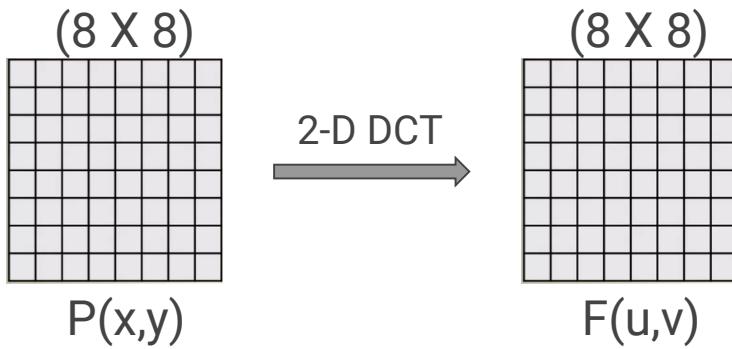
- ❖ But the image matrix is 2-D matrix. So we can either apply 1-D DCT to the image matrix twice. Once row wise, then column wise, to get the DCT coefficients. Or we can apply the standard 2-D DCT formula for JPEG compression. If the *input matrix* is  $p(x,y)$  and the *transformed matrix* is  $F(u,v)$  or  $G(u,v)$  then the DCT for the 8x8 block is computed using the expression: –
- ❖ 2-D DCT  $C(u,v) = \frac{1}{4}a(u)a(v) \sum_{x=0}^7 \sum_{y=0}^7 f(x,y) \cos \left[ \frac{(2x+1)u\pi}{16} \right] \cos \left[ \frac{(2y+1)v\pi}{16} \right]$

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# DC and AC components



$F(0,0)$  is called the ***DC Component*** and the rest of  $P(x,y)$  are called ***AC components***.

- ❖ For  $u = v = 0$  the two cosine terms are 0 and hence the value in the location  $F[0,0]$  of the transformed matrix is simply a function of the summation of all the values in the input matrix.
- ❖ This the mean of 64 values in the matrix and is known as the ***DC coefficient***.
- ❖ Since the values in all the other location of the transformed matrix have a frequency coefficient associated with them they are known as ***AC coefficients***.

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Example of DCT

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

2-D DCT



64	56	48	40	32	24	16	8
56	49	42	35	28	21	14	7
48	42	36	30	24	18	12	6
40	35	30	25	20	15	10	5
32	28	24	20	16	12	8	4
24	21	18	15	12	9	6	3
16	14	12	10	8	6	4	2
8	7	6	5	4	3	2	1

Note that this is only a hypothetical transform. Do not confuse this with DCT

# Quantization

- ❖ Humans are unable to see aspects of an image that are at really high frequencies. Since taking the DCT allows us to isolate where these high frequencies are, we can take advantage of this in choosing which values to preserve. By multiplying the DCT matrix by some mask, we can zero out elements of the matrix, thereby freeing the memory that had been representing those values.
- ❖ The resultant quantize matrix will only preserve values at the lowest frequencies up to a certain point.
- ❖ Why
  - To reduce the number of bits per sample.
- ❖ Two
  - Uniform
  - Non-uniform
  - Custom quantization tables can be put in image/scan header.
  - JPEG Standard defines two default quantization tables, one each for luminance and chrominance.

# Quantization

Standard Formula:  $\widehat{F}(u, v) = \text{round} \left( \frac{F(u, v)}{Q(u, v)} \right)$

- ❖  $F(u, v)$  represents a DCT coefficient,  $Q(u, v)$  is quantization matrix, and  $\widehat{F}(u, v)$  represents the quantized DCT coefficients to be applied for successive entropy coding.
- ❖ The quantization step is the major information losing step in JPEG compression.
- ❖ For non-uniform quantization, there are 2 psycho-visually tuned quantization tables each for luminance (Y) and chrominance (Cb, Cr) components defined by jpeg standards.

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Quantization

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

The Luminance Quantization Table

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

The Chrominance Quantization Table

- ❖ The entries of  $Q(u,v)$  tend to have larger values towards the lower right corner. This aims to introduce more loss at the higher spatial frequencies.
- ❖ The tables above show the default  $Q(u,v)$  values obtained from psychophysical studies with the goal of maximizing the compression ratio while minimizing perceptual losses in JPEG images.

AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Quantization - Example

160	80	47	39	5	3	0	0	0
65	53	48	8	5	2	0	0	0
58	34	6	4	2	0	0	0	0
40	7	6	1	1	0	0	0	0
8	4	0	0	0	0	0	0	0
5	2	0	0	0	0	0	0	0
3	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

DCT Coefficients  $F(u,v)$



Quantizer



16	7	4	2	0	0	0	0	0
5	4	3	0	0	0	0	0	0
4	2	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

Quantized Coefficients  $F(u,v)$

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Quantization Table  $Q(u,v)$

Each element of  $F(u,v)$  is divided by the corresponding element of  $Q(u,v)$  and then rounded off to the nearest integer to get the  $\hat{E}(u,v)$  matrix.

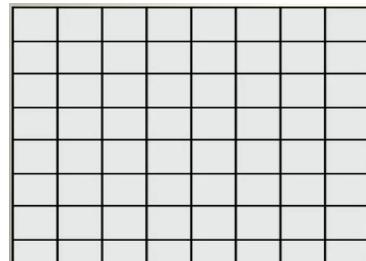
AVAILABLE AT:

Onebyzero Edu - Organized Learning, Smooth Career

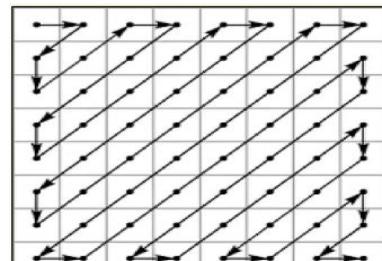
The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))

# Zig-Zag Scan

- ❖ Maps  $8 \times 8$  matrix to a  $1 \times 64$  vector.
- ❖ Why zigzag scanning?
  - To group low frequency coefficients at the top of the vector and high frequency coefficients at the bottom.
- ❖ In order to exploit the presence of the large number of zeros in the quantized matrix, a zigzag of the matrix is used.



$(8 \times 8)$



$(8 \times 8)$



$(64 \times 1)$



End Product:



$(1 \times 64)$

# Thank You

AVAILABLE AT:

**Onebyzero Edu - Organized Learning, Smooth Career**

The Comprehensive Academic Study Platform for University Students in Bangladesh ([www.onebyzeroedu.com](http://www.onebyzeroedu.com))