



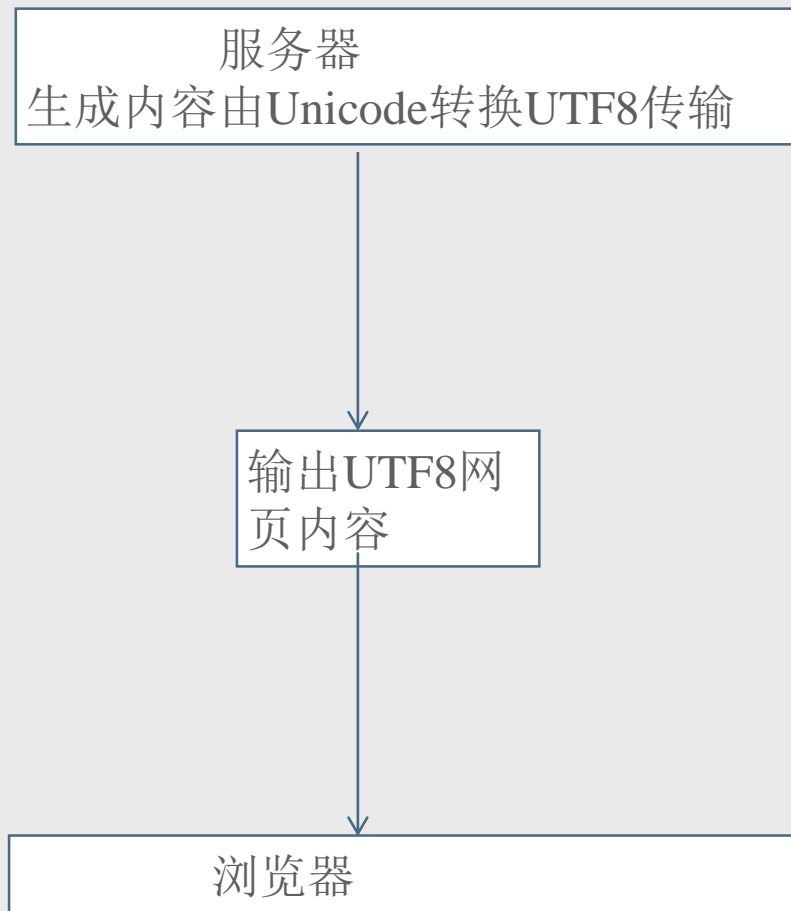
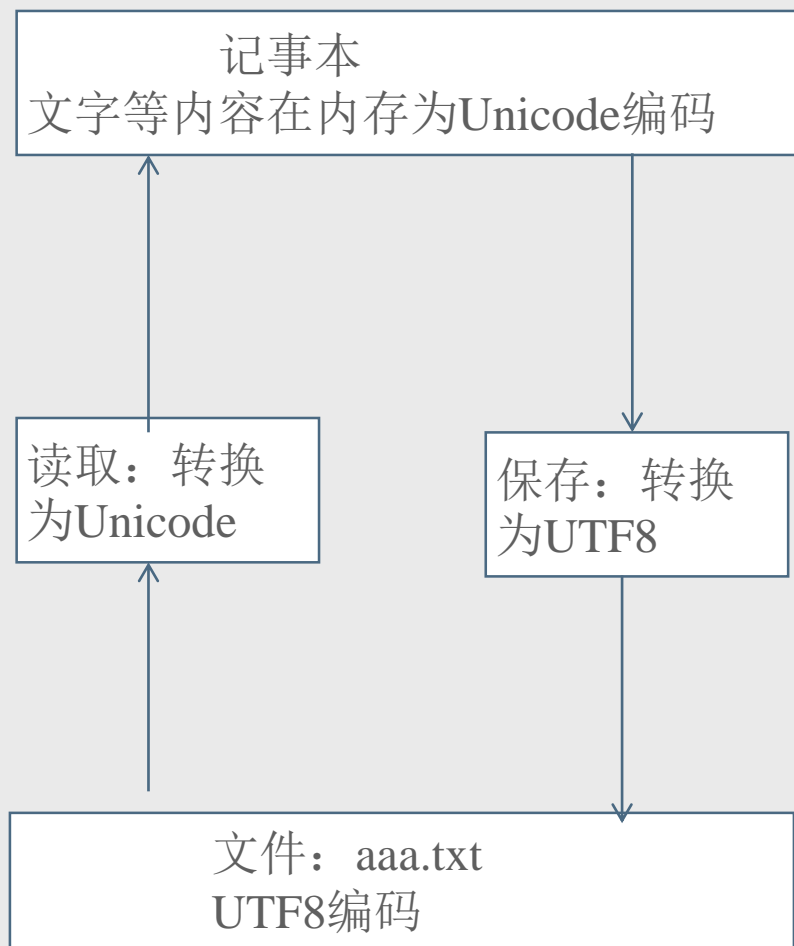
程序设计与数据结构

—— 字符编码

讲师：周宇航

1. 字符编码类型
2. 字符串编解码转化

1. ASCII编码
2. 中文GB2312
3. Unicode
4. UTF8



- 在含有中文的内容处理中，尽量使用UTF8编码
- 了解乱码的原因
- 解决乱码的手段

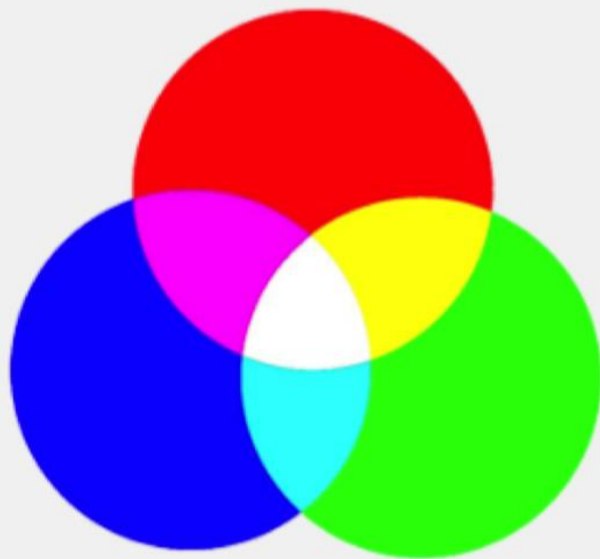
需求：

- 1.把一张图片，转化为字符画
- 2.把一个视频，转化成字符动画

原理：

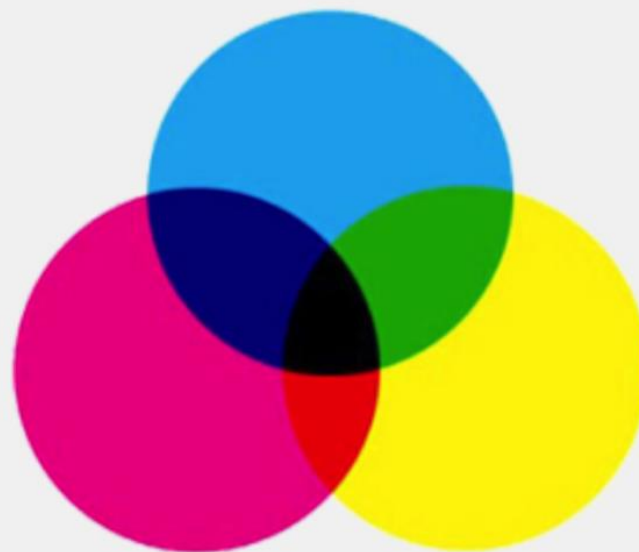
字符画是一系列字符的组合，我们可以把字符看作是比较大块的像素，一个字符能表现一种颜色（为了简化可以这么理解），字符的种类越多，可以表现的颜色也越多，图片也会更有层次感

RGB | 红色
绿色
蓝色



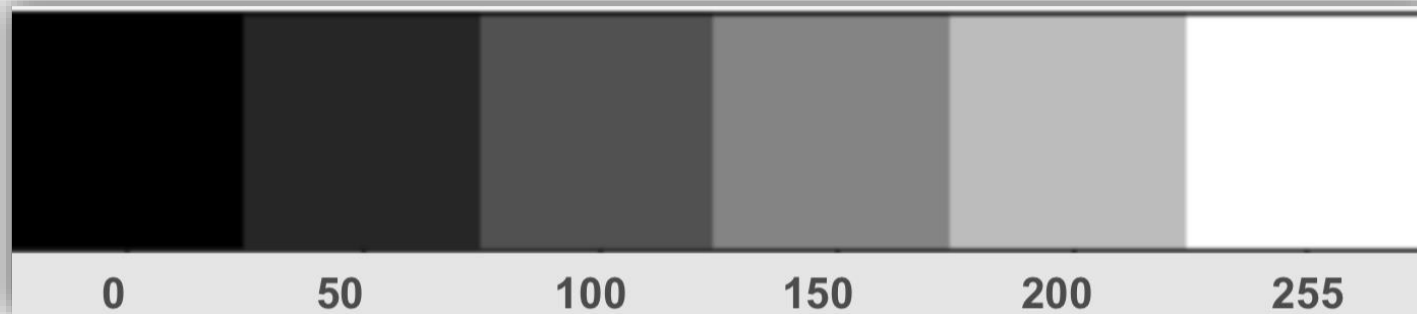
色光三原色

CMY | 青色
品红
黄色

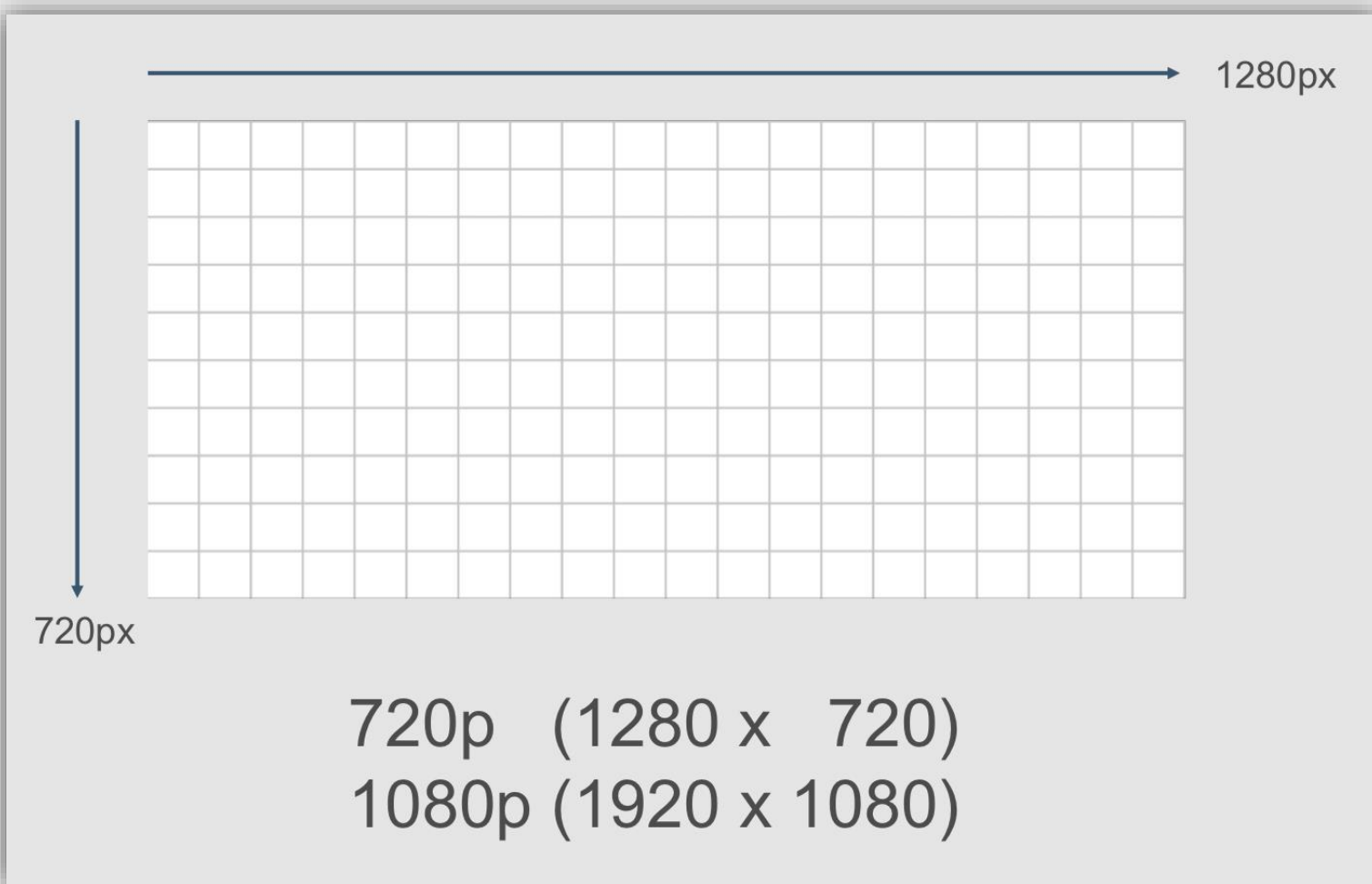


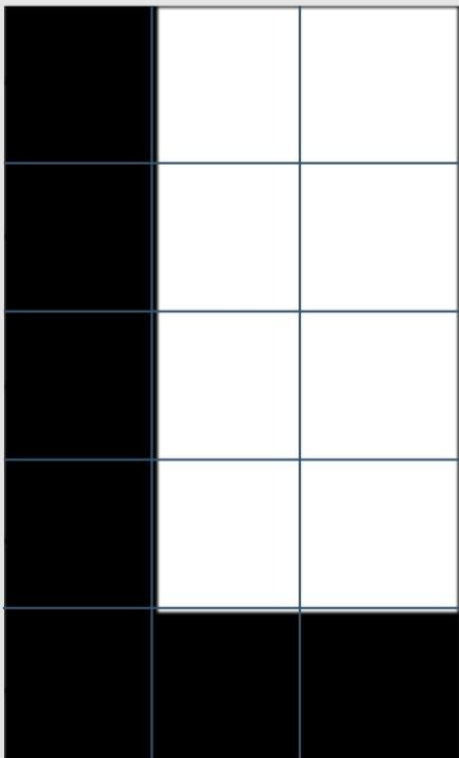
颜料三原色

	红	Red	255,0,0	#FF0000
	绿	Green	0,255,0	#00FF00
	蓝	Blue	0,0,255	#0000FF
	黄	Yellow	255,255,0	#FFFF00
	粉红	Pink	255,192,203	#FFC0CB
	深粉色	DeepPink	255,20,147	#FF1493
	草坪绿	LawnGreen	124,252,0	#7CFC00
	白	White	255,255,255	#FFFFFF
	黑	Black	0,0,0	#000000

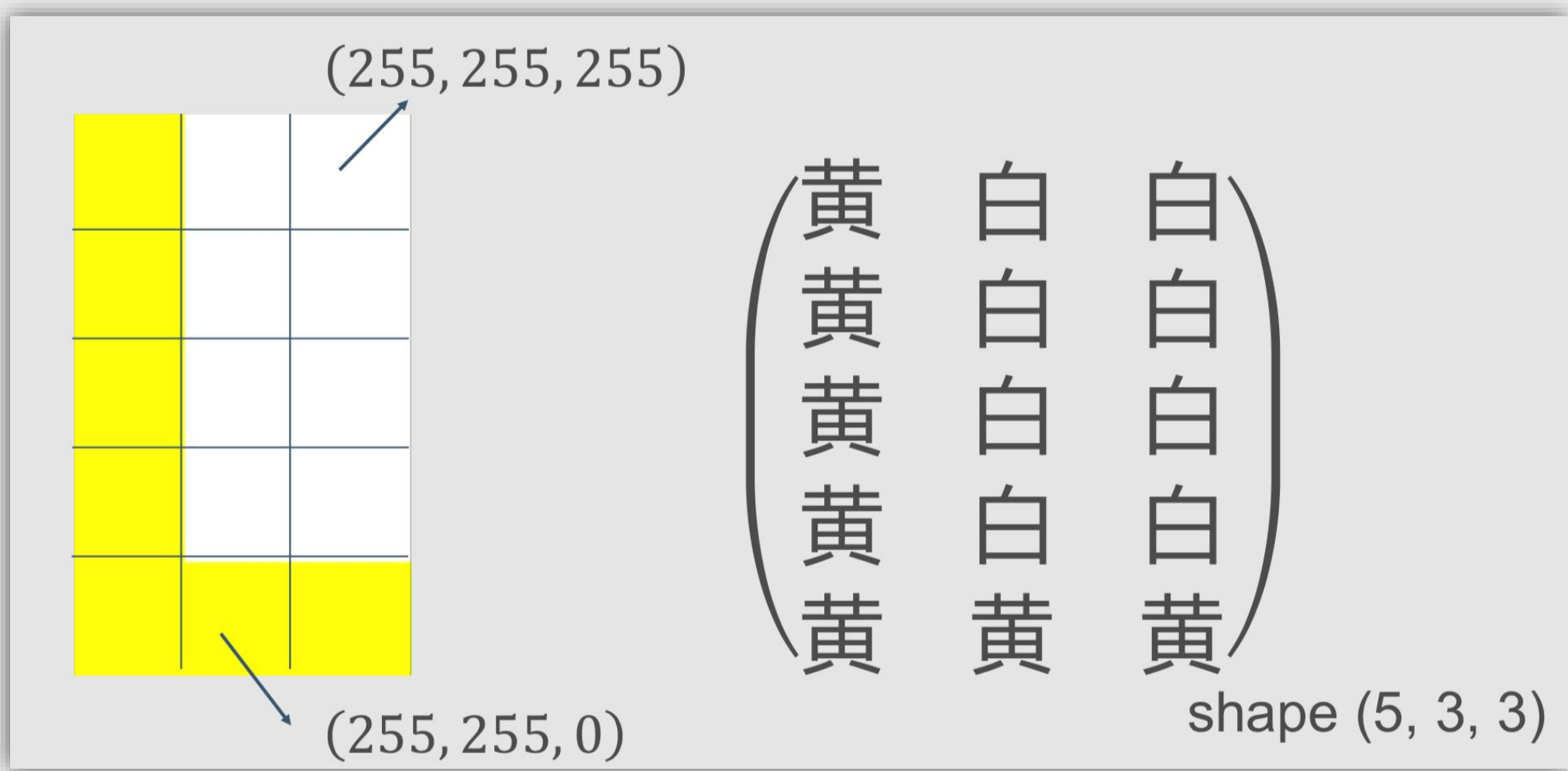







$$\begin{pmatrix} 0 & 255 & 255 \\ 0 & 255 & 255 \\ 0 & 255 & 255 \\ 0 & 255 & 255 \\ 0 & 0 & 0 \end{pmatrix}$$

shape (5, 3)



1. 首先使用 PIL 的 `Image.open` 打开图片文件，获得对象 `im`
2. 使用 PIL 库的 `im.resize()` 调整图片大小对应到输出的字符画的宽度和高度，注意这个函数第二个参数使用 `Image.NEAREST`，表示输出低质量的图片。
3. 遍历提取图片中每行的像素的 RGB 值，调用 `getchar` 转成对应的字符
4. 将所有的像素对应的字符拼接在一起成为一个字符串 `txt`
5. 打印输出字符串 `txt`
6. 如果执行时配置了输出文件，将打开文件将 `txt` 输出到文件，如果没有，则默认输出到 `output.txt` 文件

1. `pip3 install pillow`
2. `pip3 install pyprind`
3. `pip3 install numpy`
4. `pip3 install opencv-python`

CLIPPlayVideo.CharFrame

pixelToChar(self, luminance)

convert(self, img, limitSize=-1, fill=False, wrap=False)

CLIPPlayVideo.V2Char

__init__(self, path)

genCharVideo(self, filepath)

export(self, filepath)

load(self, filepath)

play(self, stream = 1)

- 注意：一定要自己动手联系，此作业不用上交

EDU

CSDN学院 IT实战派

