

第二次作业

程远 2234412848

1 第一题

对于整数部分，对 894 连续进行被除数为 2 的带余除法，记录每次计算的余数，当商为 0 时停止，将余数逆序排列，得转换结果，为 1101111110。

对于小数部分，对 145 连续进行乘数为 2 的乘法，取整数部份，并对小数部分重复操作，直到积为 0 或进入循环。将得到的积顺序排列，得转换结果，为 0.00100101（后续位数省略）。

两数相加，则 894.145 的二进制表示为 1101111110.00100101

100010111 表示十进制数 279，即 $\sum_{i=1}^k x_i 2^i$ ，其中 k 为总位数， x_i 为第 i 位上的二进制数。

2 第二题

加法运算可以通过逻辑运算实现，则加法的逆运算——减法也可以通过逻辑运算实现。乘法是加法的叠加，则也可以通过逻辑运算实现，除法是乘法的逆运算，则也可以通过逻辑运算实现，下以 6*5 为例说明。先计算 5+5，5 的二进制表示为 101，按照以下的位运算规则进行运算。

$$S_i = (A_i \text{ XOR } B_i) \text{ XOR } C_i \quad (1)$$

$$C_{i+1} = ((A_i \text{ XOR } B_i) \text{ AND } C_i) \text{ OR } (A_i \text{ AND } B_i) \quad (2)$$

其中， A_i 与 B_i 是当前位的数字， C_{i+1} 是下一位结果， S_i 是当前位结果。按照这一规则，运算 101+101 得到 1010。同理再进行四次相加，得 6*5 的二进制表示为 11110，转换为十进制即为 30。

3 第三题

十进制实数为-120.625

4 第四题

对于音频来说，把声音看作连续传播的机械波（声波），声波随时间传递和变化。每间隔一个固定时间记录时刻和声波振幅，得到以时间为横坐标，振幅为纵坐标的离散信息，即完成采样，也就是将音频对时间进行采样。

图片是对空间和时间的同时采样。对一个连续的空间来说，采样过程需要选取固定的采样面积，用灰度值或 RGB 值或其他方式来记录采样面积内的信息。当整个空间被遍历后，采样完成，得到灰度值等信息随空间变化的离散信息，这种离散信息可以被计算机处理并以图片形式呈现。对于时间来说，一张图片本身就是对某一时刻的采样。而在一段时间中，间隔定时间多次采样，变能得到连续的图片。当采样频率足够大时，人脑便会将离散的图片信息识别为连续的视频（此时采样频率等于视频帧率），实现对连续时间的采样。

5 第五题

下面介绍 JPEG 格式图片如何从图片转换为 0/1 编码，其大致分为图像分割、颜色空间转换离散余弦变换、数据量化和数据编码五个步骤。此外

图像分割，即把原始图像分割为 8×8 的小块便于后续处理，此处不多赘述。

颜色空间转换与第一步压缩：常见的图片为 RGB 颜色空间，需要将其转换为 YCbCr 空间以便压缩（这里 Y 表示亮度，Cb 和 Cr 分别表示绿色和红色的色差值），转换公式如下

$$Y = 0.299R + 0.587G + 0.114B \quad (3)$$

$$C_b = -0.1687R - 0.3313G + 0.5B \quad (4)$$

$$C_r = 0.5R - 0.4187G - 0.0813B \quad (5)$$

转换完成后，由于人类对颜色的敏感度低于对亮度的敏感度，可以利用此特点对图像进行初步压缩。具体操作为将 Cb 通道和 Cr 通道的每四个像素点用一个大像素点代替，大像素点的通道值为四个小像素点通道值的平均值，以此实现压缩。

离散余弦变换 (DCT)：一段一维函数图像可以由足够多个余弦函数图像叠加而来，此处以亮度值为例说明。利用八条基础频率曲线叠加出亮度值随像素位的曲线，本质上就将函数映射到了一个 1×8 的线性空间中。将一维扩展到二维，即可将整个 8×8 的 YCbCr 图像的每个通道分别映射到 8×8 的基础频率曲线空间中。这样我们得到了一个新的 8×8 矩阵，这个新矩阵（三通道）即为离散余弦变换的结果。

数据量化（第二步压缩）：得益于基础频率曲线的选取，DCT 后的矩阵的左上角代表原始图像中的低频信号，右下角代表原始图像中的高频信号。绝大多数图像的低频信号都远多于高频信号，体现在矩阵中即为左上角值明显大于右下角值。而又因为人类对高频信号敏感度较低，我们可以通过一定的手段丢弃高频信号实现压缩。通过将原矩阵除以 JPEG（联合图像专家组）的量化矩阵，我们能够将原矩阵右下部分变为 0，实现数据量化（第二步压缩）。

数据编码：经过以上过程，我们将一张图像变换为了若干 8×8 的子块，之后将对不同类型元素进行分别编码。第一类元素是 8×8 子块的左上角系数，称为 DC（直流分量）。由变换特性，直流分量代表的是整个子块的均值，JPEG 对这些直流分量采用差分编码，即除第一个元素外，将其中各元素都表示为各该元素与其前一元素的差的编码。这种编码能够大幅减少直流分量的重复，提高压缩效率。另一类元素是 8×8 子块的其它 63 个系数，即交流 (AC) 系数，JPEG 对其采用游程编码。为了保证这 63 个系数应该按照低频分量先出现，高频分量后出现的顺序编码（以增加行程中连续“0”的个数），JPEG 对这 63 个元素采用了“之”字型 (Zig-Zag) 的排列方法。再把所有子块的排列结果相连，就将二位图像转换为了十进制数列。为了进一步提高压缩效果，需继续对十进制数列进行哈夫曼编码，结果为一个更短的十进制序列。最后将十进制数列转为二进制数列，就完成了主体编码。再加上提前定义的文件头即可。主要的几个用于文件头的十六进制数如下：ffd8 表示 start of image，ffe0 表示 app0，即是 JFIF 格式，ffdb 表示量化表（两个 ffdb 分别表示亮度和色度分量的量化表），0043 表示量化表 65 字节（64byte 量化参数 + 1 byte 精度及量化表 ID），ffc0 表示 start of frame，即图像数据。当然，这些数字最终也会被转换为二进制数储存。

JPEG 格式的标准及其作用：JPEG 格式的标准是 JPEG 协议，JPEG 协议规范了压缩数据格式（构成部分、语法、语法图公约、符号及代码长度和值的公约等）、通用顺序和渐进语法（高层语法、帧头语法扫描头语法、表规范和各种标记字段语法等）、分层语法（高级分层模式语法、DHP 段语法、EXP

段语法)、压缩图像数据的缩写格式和表格规范数据的缩写格式等内容。当所有软件开发者、与硬件厂家都遵守同一份 JPEG 协议时,协议的作用就体现出来了。其一、兼容性大大增强。统一协议意味着不同厂家生产的设备和开发的软件之间可以相互兼容,能够正确地读取和显示 JPEG 格式的图像文件。这样的互操作性使得用户能够在不同平台之间传输 JPEG 图像。其二、提高开发效率。JPEG 协议规范了开发时的语法与算法,这大大减轻了开发者的工作量。其三、建立统一的生态。当软硬件开发者都遵守同一份协议时,一个包括图像编辑软件、摄影设备、图像存储设备、网络服务等各种相关产品和服务的生态就会逐步建立,带来便利。

6 第六题

指令编码包括对操作码和操作数进行编码。例如,在一个 8 位的指令集中,可以用 4 位编码操作码,4 位编码操作数。比方说可以用 0001 表示 ADD 指令,这里 0001 便是操作码。ADD 指令将两个寄存器 R1 和 R2 的值相加,并将结果存储到第三个寄存器 R3 中,那我们可以将 R1、R2 和 R3 依次编码为 0000、0001、0010,这便完成了一次简单的指令编码。

定长指令集是指指令长度固定的指令集,如 X86,ARM 等指令集,它们都提前指定了每条指令的长度。例如在 X86-64 指令集中每条指令的长度都是 64 位。这种指令集的优点是解码简单和执行高效,由于指令长度固定,解码器可以更容易地解析和执行指令。处理器可以以固定的步长获取指令,从而提高执行效率。这就类似线性存储相对与链式存储的优势,即可以通过下标访问元素,快捷高效。

与此相对的非定长指令集是指指令长度根据需求而变化的指令集。RISC-V 中的部分指令集便是非定长指令集。非定长指令集具有指令密度高和指令设计灵活的特点。因为指令长度可以根据需要而变化,可以实现更高的指令密度,节省存储空间。同时非定长特点使开发人员可以更灵活地设计指令集,以满足不同应用场景的需求。