

计算机二级JAVA辅导:浅谈Umd文件格式计算机等级考试

PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/645/2021_2022__E8_AE_A1_E7_AE_97_E6_9C_BA_E4_c97_645255.htm

Umd文件有两种格式，一种是文本格式，另一种是漫画格式，考虑到主要的需求还是文本格式，本文主要是基于这个格式来讲解的。首先是文件头，大部分文件都是靠文件头来区分文件格式的吧，Umd也不例外，Umd的文件头是0xde9a9b89，写到文件上前四位分别应该是0x89,0x9b,0x9a,0xde,这个大家理解起来应该没什么问题吧，以下的类似。然后第5个字节是0x23，也就是字符'#'，这个字符在Umd中被用来作为功能块的分割符。接下来有4个字节是固定字节0x01,0x00,0x00,0x08,其具体意义不明。然后是第10个字节了吧，第10个字节是决定Umd是文本还是漫画的，0x01表示是文本，0x02表示是漫画。然后是2个字节的随机数。随机数完了之后，就到了Umd的一些基本属性了，例如标题，作者等。这些属性是用同样的格式生成的，格式如下：分割符'#':1字节 属性类别:2字节，其中类别包括以下几类(标题--0x02,作者--0x03，年--0x04，月--0x05，日--0x06，小说类型--0x07，出版商--0x08，零售商--0x09) 0x00:1字节 内容长度:1字节，长度也很特别，并不是直接取字符数，而是字符数*25，例如字符串"hello",则长度是5*25=125 内容:n字节，要写入的内容是经过Unicode编码的，现在明白上面的长度为什么要乘25了吧。好了，基本属性完成之后，就开始正文内容了。首先还是分割符'#'，然后是2个字节0x0b,表示下个部分是用来记录内容长度的，(细心的读者应该已经发现了，在分割符'#'后面跟的都是数据类型，

本文在最后会给大家总结一下现在已知的数据类型)。接下来是两个字节0x0900, 然后是4个字节的内容长度(内容长度为各章正文长度之和)。长度完了之后就到了章节偏移量, 分割符'#', 数据类型2个字节0x83, 然后是2个字节的0x0901, 接下来会有4个字节的随机数, 接下来一个字节0x24, 然后又是4个字节的随机数, 这个随机数要和上一个随机数相同, 要不然生成的Umd在某些解析器下不能通过校验。接下来的4个字节和章节的数目有关, 字节内容是(章节数目*4) 9, 假如小说有两章, 那就是17, 然后, 就是每章的偏移量, 每个占4个字节, 第一章偏移当然是0啦, 第二章的偏移是第一章的正文长度*2(因为是Unicode编码)。以此类推。好, 接下来的数据类型是章节标题, 还是'#'然后接2个字节数据类型0x84, 两个字节0x0901, 4字节随机数, 0x24, 4字节随机数(需一致), 然后是4字节和小说总标题长度有关, 字节内容是(章节1标题长度*2 1) (章节2标题长度*2 1) ... 9; 然后就是写每章标题的内容了, 按如下格式写: 首先1个字节, 内容章节标题长度*2, 接下来章节标题长度*2个字节, 内容是章节长度的Unicode编码。标题写完了之后, 就要开始写正文了。注意, 正文不是按章节来写的, 而是先把所有章节的正文凑成一个长字符串, 然后分成n个数据块来写入, 并且每个数据块都使用Zlib进行压缩, 建议每个数据块按固定大小来分, 而且Umd貌似有最大数据块大小的限制, 超过了, 解析的时候会有问题, 建议数据块大小32768字节。下面说明如何写数据块, 第一个字节是0x24(注意, 不是'#'), 然后是4个字节随机数, 然后是4个字节, 内容是压缩后数据块长度 9, 然后就把压缩后数据块写入就完成了了一个数据块

的写入，在写完每个数据块后，可以选择做如下两件事的一件或两件（建议用随机数来决定）：1. 写入1个字节'#'，2个字节的0xf1,2个字节的0x1500，16个字节的0x02。2. 写入1个字节'#'，2个字节的0x0a,2个字节的0x0900，4个字节的随机数。在所有正文数据块写入完毕后，写入1个字节'#'，2个字节数据类型0x81，表示正文写入完毕，2个字节0x0901，4个字节随机数，1个字节0x24，4个字节随机数（一致），接下来是4个字节，内容是数据块的数目*49，然后，还记得每个数据块写入前都生成了4个字节的随机数吗，从最后一个开始，倒序写如这些随机数，每个4个字节，结束正文的写入。接下来是封面，1个字节的'#'，2个字节的0x82,3个字节0x010a01，4个字节的随机数，1个字节的0x24，4个字节的随机数，4个字节的0x83，内容和封面字节数有关，内容是封面字节数*9，然后再写入封面字节数据，不需要压缩。到此为止，主要数据都写完了，但我发现有些Umd生成工具还会生成一些数据类型是0x87的，这一块具体功能还不是非常清楚，大概是生成一些数据来通知Umd解析器应该使用什么样的PageOffset来显示内容，具我测试，这部分不写入也能正常解析，对这部分有兴趣的朋友，可以参考我提供的代码。该结束了，1个字节的'#'，2个字节的0x0c表示文件结束，2个字节的0x0901,4个字节的文件长度4，完成了。附，数据类型表：0x01--文件开始 0x02--标题 0x03--作者 0x04--年 0x05--月 0x06--日 0x07--小说类型 0x08--出版商 0x09--零售商 0x0b--内容长度 0x83--章节偏移 0x84--章节标题，正文 0x81--正文写入完毕 0x82--封面 0x87--PageOffset 0x0c--文件结束。写了一大堆，写完之后才发现像流水帐，看来和电脑用长

了文字功底都还老师了，不过应该还是能看明白的^_^。编辑特别推荐: 计算机等级考试二级JAVA考前密卷及参考答案 全国计算机等级考试二级JAVA全真模拟试题 全国计算机等级考试二级JAVA在线模拟考试 计算机等级考试二级Java初级练习(精选25题) JAVA技术学习法 Java学习如何走出第一步 最值得学习的五种JAVA技术 100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com