Base64 编码

给定一个文件,输出这个文件的 Base64 编码。

Base64 编码规则如下:

- 1. 将每三个字节作为一组,一共是 24 个二进制位。
- 2. 将这 24 个二进制位分为四组,每个组有 6 个二进制位。
- 3. 在每组前面加两个00,扩展成32个二进制位,即四个字节。
- 4. 根据下表,得到扩展后的每个字节的对应符号,这就是 Base64 的编码值。

索引	编码	索引	编码	索引	编码	索引	编码
0	A	16	Q	32	g	48	W
1	В	17	R	33	h	49	Х
2	С	18	S	34	i	50	У
3	D	19	Т	35	j	51	Z
4	Е	20	U	36	k	52	0
5	F	21	V	37	I	53	1
6	G	22	W	38	m	54	2
7	Н	23	X	39	n	55	3
8	I	24	Υ	40	0	56	4
9	J	25	Z	41	р	57	5
10	K	26	а	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	С	44	S	60	8
13	N	29	d	45	t	61	9
14	0	30	е	46	u	62	+
15	Р	31	f	47	V	63	1

如果字节数不足三,则这样处理:

- 两个字节的情况:将这两个字节的共 16 个二进制位按照上面的规则转成三组,最后一组除了前面加两个 0 以外,后面再加 2 个 0。这样得到一个三位的 Base64 编码,再在末尾补上 1 个 🖃 号。
 - 。 比如,"Ma" 这个字符串是两个字节,可以转化成三组 00010011、00010110、00010000 以后,对应 Base64 值分别为 T、W、E,再补上一个 号,因此 "Ma" 的 Base64 编码就是 TWE■。
- 一个字节的情况:将这一个字节的 8 个二进制位,按照上面的规则转成两组,最后一组除了前面加两个 0 以外,后面再加 4 个 0。这样得到一个两位的 Base64 编码,再在末尾补上 2 个 🗐 号。
 - 比如,"M" 这个字符串是一个字节,可以转化为两组 00010011、00010000,对应的 Base64 值分别为 T、Q,再补上两个 = 号,因此 "M" 的 Base64 编码就是 TQ==。

样例:

编码前	1							2								3								
位 模 式	0	0	1	1	0	0	0	1	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	1
索引	12										8						51							
编码后	М										I				z									

输入描述

我们会提供一些文件,向你的程序输入文件名,你需要读取指定的文件计算其 Base64 编码。

输出描述

输出指定文件的 Base64 编码。如果文件不存在,输出 FILE NOT FOUND。

示例 1

提供文件 numbers.txt , 内容为 0123456789

1 输入: numbers.txt 2 输出: MDEyMzQ1Njc40Q==

示例 2

提供文件 hello.c

1 输入: not-exists 2 输出: FILE NOT FOUND

提示

- 使用位运算
- 输入的文件可能是各种格式的,你需要使用 ios_base::binary 打开文件
- 注意: 读文件时请使用绝对路径, 使用相对路径可能会出问题