A-Level Computer Science

Formula Sheet · 公式表

*9618 supplies no formula sheet, and most of what is needed is not a formula at all but a **range or a size** that has to be known cold: what an 8-bit two's complement number can hold, how many bits an image takes, how many comparisons a binary search needs. Every one of them is a mark on its own, and each is a single line to learn. 9618 不提供公式表; 以下大多不是 " 公式 ", 而是必须熟记的取值范围与容量计算。*

Number bases

Binary place values 二进制位权 — for one byte, 128 64 32 16 8 4 2 1 from left to right. Write the row down first and conversion becomes addition.

Hexadecimal place values 十六进制位权 — . . . 256 16 1. Each hex digit is exactly FOUR bits, which is the whole reason hex is used: one byte is always two hex digits.

Denary to binary 十进制转二进制 — repeatedly divide by 2 and read the remainders upwards, or subtract place values from the largest down. The two methods should agree – use the second as a check.

Binary to hexadecimal 二进制转十六进制 — split into groups of four bits FROM THE RIGHT, and convert each group. Padding with leading zeros on the left is safe; padding on the right is not.

Binary coded decimal (BCD) 二进制编码的十进制 — each denary digit stored in its own four bits, so 47 is 0100 0111. It is not the same as the binary for 47, and the point of it is exact decimal arithmetic for money.

Ranges – know these without working them out

Unsigned integer range 无符号整数范围 — n bits hold 0 to $2^n - 1$. For 8 bits that is 0 to 255; for 16 bits, 0 to 65 535.

Two's complement range 补码范围 — n bits hold -2^{n-1} to $2^{n-1} - 1$. For 8 bits that is −128 to +127. The range is NOT symmetric: there is one more negative value than positive, because zero takes up a positive-looking pattern.

Forming a two's complement negative 求补码 — write the positive number in binary, invert every bit, then add 1. So +53 = 0011 0101 becomes 1100 1011 = −53.

Reading a two's complement number 读补码 — if the leftmost bit is 1 the number is negative: invert and add 1 to find its size. The leftmost bit is a SIGN indicator, not a place value of 128.

Number of distinct values 可表示的数值个数 — n bits give 2^n different patterns, whether the number is signed or not. Signing changes which values, never how many.

Overflow 溢出 — the result needs more bits than are available. In two's complement addition, overflow has occurred when adding two positives gives a negative, or two negatives give a positive.

Shifts, floating point and Boolean algebra

Logical shift left by n 逻辑左移 — multiplies an unsigned value by 2^n, and bits shifted off the left are lost. 0000 0110 shifted left twice is 0001 1000: $6 \times 4 = 24$.

Logical shift right by n 逻辑右移 — divides by 2^n, discarding the remainder. Zeros are shifted in on the left.

Arithmetic shift right 算术右移 — copies the SIGN bit in on the left instead of a zero, so a negative two's complement number stays negative. This is the one a signed division needs.

Floating point value 浮点数的值 — value = mantissa $\times 2^{\text{exponent}}$, with both parts stored in two's complement. So mantissa 0.101 with exponent 011 is $0.625 \times 2^3 = 5$.

Normalised form 规格化 — the first two bits of the mantissa must DIFFER: 01 . . . for a positive number, 10 . . . for a negative one. Normalising keeps the greatest precision for the bits available.

De Morgan's laws 德摩根定律 — $\overline{A \cdot B} = \bar{A} + \bar{B}$ and $\overline{A + B} = \bar{A} \cdot \bar{B}$. Break the bar and change the sign – the phrase is the mnemonic and the method.

How much space data takes

Bitmap image size 位图文件大小 — width in pixels × height in pixels × colour depth in bits. That gives BITS; divide by 8 for bytes. Add the header if the question mentions one.

Number of colours 颜色数 — $2^{\text{colour depth}}$. A colour depth of 8 bits gives 256 colours; 24 bits gives about 16.7 million.

Sound file size 音频文件大小 — sample rate (Hz) × sample resolution (bits) × duration (s) × number of channels. Again the result is in bits.

Bit rate 比特率 — sample rate × sample resolution × channels, in bits per second. File size is then bit rate × duration.

Transmission time 传输时间 — file size ÷ transmission rate, with BOTH in the same unit. A file in bytes and a link in bits per second differ by a factor of 8 – convert before dividing.

Compression ratio 压缩比 — compressed size ÷ original size, usually as a percentage. "Reduced BY 60%" and "reduced TO 60%" are different claims; read which one the question wants.

Data units 数据单位 — 1 KiB = 1024 bytes, 1 MiB = 1024 KiB, 1 GiB = 1024 MiB – while 1 kB = 1000 bytes. Use whichever the question uses and say which.

An image is 800 by 600 pixels with a colour depth of 24 bits. Find its size in MiB.

About 1.37 MiB (11\,520\,000 bits, or 1\,440\,000 bytes).

$Bits = 800 \times 600 \times 24 = 11\,520\,000$

$Bytes = 11\,520\,000 \div 8 = 1\,440\,000$

$KiB = 1\,440\,000 \div 1024 = 1406.25$

$MiB = 1406.25 \div 1024 = 1.37$

A one-minute stereo recording is sampled at 44.1 kHz with a resolution of 16 bits. Find its

size in MiB.

About 10.1 MiB.

$Bits = 44\,100 \times 16 \times 2 \times 60 = 84\,672\,000$

$Bytes = 84\,672\,000 \div 8 = 10\,584\,000$

$MiB = 10\,584\,000 \div 1024 \div 1024 = 10.1$

Text, addresses and error checking

ASCII ASCII — 7 bits per character, so 128 different characters; extended ASCII uses 8 bits for 256. One character of ASCII text is stored in one byte.

Unicode Unicode — a much larger character set covering every writing system, at the cost of more bits per character. The trade-off – more characters, bigger files – is the marked point.

IP addresses IP 地址 — IPv4 is 32 bits, written as four denary octets each 0 to 255; IPv6 is 128 bits, written as eight groups of four hex digits.

MAC address MAC 地址 — 48 bits, written as six pairs of hex digits. The first half identifies the manufacturer, the second the device.

Parity check 奇偶校验 — one bit is set so the number of 1s is even (even parity) or odd (odd parity). It detects an odd number of flipped bits and misses an even number – that limitation is the exam answer.

Checksum 校验和 — a value calculated from the data, sent with it and recalculated on arrival. Different value, corrupted data – but it says nothing about WHERE the error is.

Check digit 校验位 — an extra digit computed from the others, as in ISBN and barcodes. It catches mistyped and transposed digits at the point of entry.

Algorithm efficiency

Linear search 线性查找 — $O(n)$: on average $n/2$ comparisons, at worst n. It needs no ordering at all, which is its one advantage.

Binary search 二分查找 — $O(\log n)$: the maximum number of comparisons is $\lceil \log_2(n+1) \rceil$. A list of 1000 needs at most 10, and a list of a million at most 20 – but the list MUST be sorted first.

Bubble sort and insertion sort 冒泡排序与插入排序 — $O(n^2)$ in the worst case. Bubble sort makes $n-1$ passes over n items, and the number of comparisons in a full pass falls by one each time.

Merge sort and quicksort 归并排序与快速排序 — $O(n \log n)$ on average. Merge sort keeps that in the worst case too, at the price of extra memory.

Huffman coding, total bits 哈夫曼编码总位数 — $\sum$ (frequency of a symbol $\times$ length of its code). Compare with $n \times 8$ for plain ASCII to state the saving.

Big-O describes how the work GROWS with n, not how long anything takes. “$O(\log n)$ is faster” is only true for large n, and a question about a list of five items may well have a different answer.

大 O 表示增长趋势, 不等于实际运行时间。

Where these marks are lost

Reading the leftmost bit of a two’s complement number as 128. It carries −128, which is why 1000 0000 is −128 and not +128.

补码最高位的位权是 −128。

Quoting the two’s complement range as −127 to +127. It is −128 to +127, and the asymmetry is itself an exam answer.

8 位补码范围是 −128 到 +127, 不对称。

Leaving a file size in bits when the question asked for bytes, or dividing by 1000 where the question used KiB. Write the unit at every line of the working, not just the last.

每一步都写单位, 注意 bit/byte 与 1000/1024 的区别。

Grouping bits into fours from the LEFT when converting to hex. Group from the right and pad the left.

二进制转十六进制要从右往左每四位一组。

Using binary search on an unsorted list. Half the marks in such a question are for saying it must be sorted first.

二分查找的前提是列表已排序。