

起始区块	阶段	比特币/区块	年	阶段产量	阶段结束总量	已产占比
0	1	50.00000000	2009.007	10500000.00000000	10500000.00000000	50.00000006%
210000	2	25.00000000	2013.000	5250000.00000000	15750000.00000000	75.00000008%
420000	3	12.50000000	2016.993	2625000.00000000	18375000.00000000	87.50000010%
630000	4	6.25000000	2020.986	1312500.00000000	19687500.00000000	93.75000010%
840000	5	3.12500000	2024.978	656250.00000000	20343750.00000000	96.87500011%
1050000	6	1.56250000	2028.971	328125.00000000	20671875.00000000	98.43750011%
1260000	7	0.78125000	2032.964	164062.50000000	20835937.50000000	99.21875011%
1470000	8	0.39062500	2036.956	82031.25000000	20917968.75000000	99.60937511%
1680000	9	0.19531250	2040.949	41015.62500000	20958984.37500000	99.80468761%
1890000	10	0.09765625	2044.942	20507.81250000	20979492.18750000	99.90234386%
2100000	11	0.04882812	2048.934	10253.90620000	20989746.09270000	99.95117198%
2310000	12	0.02441406	2052.927	5126.95260000	20994873.04530000	99.97558804%
2520000	13	0.01220703	2056.920	2563.47630000	20997436.52160000	99.98779307%
2730000	14	0.00610351	2060.913	1281.73710000	20998718.25870000	99.99389658%
2940000	15	0.00305175	2064.905	640.86750000	20999359.12620000	99.99694833%
3150000	16	0.00152587	2068.898	320.43270000	20999679.55890000	99.99847420%
3360000	17	0.00076293	2072.891	160.21530000	20999839.77420000	99.99923713%
3570000	18	0.00038146	2076.883	80.10660000	20999919.88080001	99.99961859%
3780000	19	0.00019073	2080.876	40.05330000	20999959.93410001	99.99980932%
3990000	20	0.00009536	2084.869	20.02560000	20999979.95970001	99.99990468%
4200000	21	0.00004768	2088.861	10.01280000	20999989.97250001	99.99995236%
4410000	22	0.00002384	2092.854	5.00640000	20999994.97890001	99.99997620%
4620000	23	0.00001192	2096.847	2.50320000	20999997.48210001	99.99998812%
4830000	24	0.00000596	2100.840	1.25160000	20999998.73370001	99.99999408%
5040000	25	0.00000298	2104.832	0.62580000	20999999.35950001	99.99999706%
5250000	26	0.00000149	2108.825	0.31290000	20999999.67240001	99.99999855%
5460000	27	0.00000074	2112.818	0.15540000	20999999.82780001	99.99999929%
5670000	28	0.00000037	2116.810	0.07770000	20999999.90550001	99.99999966%
5880000	29	0.00000018	2120.803	0.03780000	20999999.94330001	99.99999984%
6090000	30	0.00000009	2124.796	0.01890000	20999999.96220000	99.99999993%
6300000	31	0.00000004	2128.788	0.00940000	20999999.97060001	99.99999997%
6510000	32	0.00000002	2132.781	0.00420000	20999999.97480001	99.99999999%
6720000	33	0.00000001	2136.774	0.00210000	20999999.97690000	100.00000000%
6930000	34	0.00000000	2140.767	0.00000000	20999999.97690000	100.00000000%

下面挑几个重点分析一下这张表。

【2】 50.00000000

格林威治时间2009年1月3日18:15:05，创世区块诞生。创世区块的编号是0。从创世区块开始的“阶段1”，每个区块产生50个新的比特币或者说50亿聪。

创世区块：<https://blockchain.info/block-height/0>



【4】 4年1次的约定

每4年减半是不太严格的说法。实际情况：比特币大约每10分钟产生一个区块，而21000个10分钟接近4年（4年等于210384个10分钟。这应该是中本聪特意选取的数字）。

【5】 2016前，2016后

2016年将发生第二次减半，但现在讨论这个有点早。我要说的是2016个块的问题。

比特币系统调节挖矿难度的原理是：根据前2016个块产生的总时间，调整后2016个块的挖矿难度，让挖出这2016个块的时间为14天。因为，每小时6个10分钟乘以24小时再乘以14天=2016。所以，所谓10分钟只是平均值目标。由于目前算力上涨很快，实际上挖出2016个块的速度往往少于14天。

难度调整的话题涉及到挖矿，以后再一并分析。

贰 选2100万的真正原因

网络上有很多种猜测，有些很靠谱，有些不靠谱但很欢乐。



【答案3】

He chose a reward scheme and 10 minute blocks. When he did the math, it came to 21 million. He didn't choose the 21 million, he just accepted the consequence of the parameters he chose.

翻译：中本聪订好10分钟、50币、4年减半的原则，结果自然出来了。他没有选，而是接受了这个自然的结果。

这个答案也是有可能的。中本聪在比特币中的很多选择确实是撞大运的，但都是“基于经验的撞大运”。

【答案4】

All gold mined in human history can be fit into a cube roughly 21 meters on each side.

Satoshi created bitcoin with the idea of being sort of a digital analog of

gold (finite supply, mining, etc), as well as the fact that it built upon Nick Szabo' s "Bit Gold" proposal, so I think that 21 million was sort of a clever nod to that.

翻译：全世界所有黄金熔在一起，是一个边长大约为21米的正方体。中本聪用这个概念，隐喻比特币是一种虚拟黄金。

原来阴谋论不止中国有...



详细：

比特币有争议的属性之一就是它的固定的供应量。当前每10分钟又25个新的比特币被生产出来，并且这一数字每4年减半。

总的来讲，不会有超过2100万个比特币的存在。另一方面，每个比特币可以被划分成1亿份（每份叫做1“聪”），如果一美分都足够买辆车的话，用美元来交易就麻烦重重了，但比特币就算升值到和上面假设的美元的状况，也不会遇到那样的问题。因此，总之，将永远存在的货币单位的总数字是2,100,000,000,000,000，也就是2100万亿，或者说250.899。在选择这个数值的方面，中本聪比大多数人意识到的要幸运的多或者说聪明的多。

首先，这个数字远小于 $2^{64}-1$ ，这是一台计算机里面可以以标准整数形式存放的最大整数，超过那个值的话， $>$ 数值将像里程表那样归零。

其次，然而，还有一个总“聪”数要设法低于的更小的阈值：可以用浮点的格式表示的可能的最大整数。整数不是计算机可以存储的唯一一种数字；为了处理小数，计算机 $>$ 使用一种做浮点表示法的格式。浮点表示法本质上就是一个科学记数法的二进制版本。

举个例子，下面是一个在你学习物理学的时候会遇到的值：

地球的质量: 5.972 1024 kg

太阳的质量: 1.989 1030 kg

光速: 2.998 108 m/s

一光年: 9.460 1015 m

质子的质量: 1.672 10⁻²⁷ kg

普朗克长度: 1.616 10⁻³⁵ m

我们可以注意到，科学记数法是如何使得你可以在合理的精度下表示所有的这些数值，尽管它们的大小相差极大。浮点表示法本质上就是二进制的科学记数法；当你存储数字9.625的时候，你的计算机存放的是“1.001101 * 10¹¹”（或者说，它存放的是01000000 00100011 01000000 00000000 00000000 00000000 $>$ 00000000 00000000，这是高精度序列形式的同样一回事）。在这个高精度形式中，系数（也就是不是指数的那部分）有52位（52bits）。

这意味着高精度（更加精确的说法是“双精度”）浮点数足以存储高达 2^{53} 的数字，但不能再高了，如果超过了，你就得开始砍掉末尾的数字。比特币的250.9这一以指数形式表现的总“聪”数，刚好低于这个最大值。

如果我们有了整数，我们为什么还要关心浮点值呢？因为更多的高阶编程语言（比如说Javascript）并不开放低阶的“浮点”和“整数表示法”，而只给程序员提供“数”的概念——当然以浮点的形式提供。如果中本聪当时选择了2亿1千万而不是2100万这个值的话，用很多语言里比特币编程就会比现在要麻烦得多了。

注意，Stefan Thomas不幸的在他写BitcoinJS的时候没有及时留意到这个，以至于

那个库使用了一个专门的 ‘大数big number’ 对象，而不是一个普通数来存储教程输出值；我自己分叉的的BitcoinJS（同时还加入了其他的改进）使用了普通数。